



ÉCOLE
POLYTECHNIQUE
DE BRUXELLES



UNIVERSITÉ LIBRE DE BRUXELLES

From discrete- to continuous-variable protocols for quantum key distribution

Mémoire présenté en vue de l'obtention du diplôme
d'Ingénieur Civil Physicien à finalité spécialisée

Célia Griffet

Directeur

Professeur Nicolas Cerf

Superviseur

Zacharie Van Herstraeten

Service

QuIC

Année académique

2018 - 2019

Abstract

"From discrete- to continuous-variable protocols for quantum key distribution" by Célia Griffet, Université Libre de Bruxelles, 2018-2019.

Quantum cryptography goes back to the mid 80s, when the first protocol for quantum key distribution was invented by Bennett and Brassard: the seminal BB84 protocol. This protocol is based on discrete-information carriers (a dichotomic state of a single photon encodes each secret key bit) and it requires single-photon detectors. Since its inception, it has been demonstrated within several experimental platforms and has been improved in various ways. For example, cryptographic keys have been distributed through optical fibers over hundreds of kilometres using a time-bin discrete encoding and alternative protocols have been devised that are more robust to line losses than BB84. Of particular relevance to this Ms thesis, alternative protocols have been developed based on continuous-information carriers (i.e., the quadrature components of the light field) and coherent measurements (i.e., homodyne or heterodyne detection) instead of photodetectors, thereby significantly enhancing the achievable bandwidth. The first such protocol was invented at QuIC in 2001 and can be viewed as the continuous analogue of BB84. The objective of this Ms thesis was to explore whether the specific techniques that had been developed in the context of continuous-variable protocols may be valuable when translated back to discrete-variable protocols. In particular, it is recognized that a “reverse” reconciliation procedure often yields higher secret key rates in continuous-variable protocols than the usual “direct” reconciliation procedure that is common to all discrete-variable protocols. The approach pursued in this Ms thesis was to map a class of known continuous-variable protocols into discrete-variable protocols going beyond BB84, and check especially whether reverse reconciliation provides an advantage here too. The security of these protocols against individual attacks could be studied by exploiting the Csiszár-Körner theorem, giving a bound on the secret key rate. Dealing with reverse reconciliation for discrete-variable protocols required developing a time-reversed version of the optimal phase-covariant quantum cloning transformation, which had not been previously described in the literature. This made it possible to characterize eavesdropping in case the secret key is built from the bits measured by the receiver instead of those generated by the sender. Numerical simulations were carried out in order to compare the achievable key rates of the new discrete-variable protocols and revealed that, against expectation, direct reconciliation-based BB84 cannot be outperformed. Finally, possible explanations of this discrepancy between discrete- to continuous-variable protocols were provided.

Key words: Quantum key distribution, BB84 protocol, Shannon information theory, Csiszár-Körner theorem, quantum cloning transformation, direct and reverse reconciliation.

Résumé

"From discrete- to continuous-variable protocols for quantum key distribution" by Célia Griffet, Université Libre de Bruxelles, 2018-2019.

La cryptographie quantique remonte au milieu des années 80, lorsque le premier protocole de distribution de clé quantique a été inventé par Bennett et Brassard : le protocole BB84. Ce protocole est basé sur des supports d'information discrets (chaque bit de la clé secrète est encodé dans l'état binaire d'un photon) et nécessite des détecteurs de photons uniques. Depuis sa création, ce protocole a été démontré sur plusieurs dispositifs expérimentaux et a été amélioré de diverses façons. Par exemple, des clés de codage ont été distribuées à travers des fibres optiques sur des centaines de kilomètres à l'aide de photons encodés discrètement et d'autres protocoles ont été mis au point qui sont plus robustes aux pertes dans les lignes que BB84. D'autres protocoles alternatifs ayant une grande importance pour ce mémoire ont par la suite été développés à base de supports d'information continus (c'est-à-dire des quadratures du champ lumineux) et de mesures cohérentes (c'est-à-dire la détection homodyne ou hétérodyne) au lieu des photodétecteurs. Cela a permis d'améliorer considérablement la bande passante atteignable. Le premier protocole de ce type a été inventé au QuIC en 2001 et peut être considéré comme l'analogue continu du protocole BB84. L'objectif de cette thèse était d'étudier si les techniques spécifiques qui avaient été développées dans le contexte des protocoles à variables continues peuvent être utiles lorsqu'elles sont traduites en protocoles à variables discrètes. En particulier, il est connu qu'une procédure de réconciliation " inverse " donne souvent lieu à des taux de clés secrètes plus élevés dans les protocoles à variables continues que la procédure de réconciliation " directe " habituelle qui est commune à tous les protocoles à variables discrètes. L'approche suivie dans cette thèse a été de faire correspondre à chaque type de protocoles à variables continues connus un protocole à variables discrètes allant au-delà de BB84, et en particulier, de vérifier si la réconciliation inverse offre un avantage ici aussi. La sécurité de ces protocoles contre les attaques individuelles peut être étudiée en exploitant le théorème de Csiszár-Körner qui donne une limite au taux de clé secrète. Le traitement de la réconciliation inverse pour les protocoles à variables discrètes a nécessité la mise au point d'une version alternative de la transformation optimale de clonage quantique à covariance de phase où le temps est inversé ; ce qui n'avait jamais été décrit auparavant dans la littérature. Cela a permis de caractériser l'espionnage dans le cas où la clé secrète serait construite à partir des bits mesurés par le récepteur au lieu de ceux générés par l'émetteur. Des simulations numériques ont été effectuées afin de comparer les taux de clés réalisables avec les nouveaux protocoles à variables discrètes et ont révélé que, contrairement à ce qui était attendu, il n'est pas possible d'avoir des performances meilleures que celles du protocole BB84 basé sur la réconciliation directe. Enfin, des explications possibles de cet écart entre les protocoles à variables discrètes et les protocoles à variables continues ont été fournies.

Mots clés : Distribution de clé quantique, protocole BB84, théorie de l'information de Shannon, théorème de Csiszár-Körner, clonage quantique, réconciliation directe et inverse.

Remerciements

Tout d'abord, je tiens à remercier Monsieur Cerf qui m'a permis de réaliser ce mémoire et a pris de son temps pour me guider et m'aider dans ma recherche.

Je remercie également Zacharie Van Herstraeten qui était présent à toutes les réunions et à chaque fois que j'ai rencontré un problème lors de la réalisation de ce mémoire.

Je tiens aussi à remercier Alfred qui a eu la patience de relire tout mon mémoire afin de voir si tout était lisible.

Je remercie également tous les membres du QuIC qui m'ont chaleureusement accueillie lors de mon arrivée au QuIC ainsi que les personnes ayant travaillé avec moi au service dont Philippe Neuville.

Finalement, je tiens à remercier ma famille, mes amis et mon copain qui m'ont soutenue lors de toutes mes années d'étude.

Contents

Introduction	4
I Background and state of the art for quantum key distribution	6
1 Basics of quantum mechanics	7
1.1 Definition of a qubit	7
1.2 Density matrix	8
1.3 Composite systems and quantum entanglement	9
1.3.1 Bases for two qubits and entanglement	9
1.3.2 Tensor product	10
1.4 Trace and partial trace	11
1.5 Measurements	11
1.5.1 Projective measurement	12
1.5.2 POVM (positive-operator valued measure) measurement	13
1.6 Conclusion	14
2 Basics of information theory	15
2.1 Shannon entropy	15
2.2 Joint and conditional entropy	16
2.3 Mutual information and its interpretation	17
2.4 Csiszar-Körner theorem for the distillation of a secret key	18
3 Quantum cryptography protocols	19
3.1 Introduction to quantum cryptography: origins and objective	19
3.2 Basic principle	20
3.3 Classification of possible attacks	22
3.4 Discrete variables protocols	22
3.4.1 BB84	23
3.4.2 Other protocols	27
3.5 Continuous variables protocols	28
3.5.1 Four canonical protocols	28
3.5.2 Equivalent entanglement-based protocols	31
3.6 Conclusion	32
4 Cloning of a quantum state	33
4.1 No-cloning theorem	33
4.2 Imperfect cloning	35
4.2.1 Universal cloning machine	35
4.2.2 Phase-covariant cloning	36

4.3	Conclusion	38
II	Results	39
5	Objective of the research	40
6	Security of the four canonical protocols supplemented with direct reconciliation	41
6.1	Introduction	41
6.2	Description of the protocols	41
6.2.1	States sent by Alice	41
6.2.2	Measurement done by Bob	43
6.2.3	Action of Eve	43
6.3	Mathematical description of the different stages	45
6.4	Results and security of the different protocols	48
6.4.1	Calculation of the mutual information	48
6.4.2	Graphs and analysis	50
6.5	Conclusion	53
7	Security of the four canonical protocols supplemented with reverse reconciliation	54
7.1	Cloning machine and protocol	54
7.1.1	Study of the cloning machine used by Eve	54
7.1.2	Calculation of the joint probability of measurements	57
7.1.3	Fidelity between Alice and Bob	59
7.2	Analysis	60
7.3	Conclusion	61
8	Discussion and conclusion	62
A	Codes for the numerical simulations	67
A.1	Protocols with direct reconciliation	67
A.1.1	Calculation of the probability with Mathematica	67
A.1.2	Analysis of the different protocols	70
A.1.3	Evolution of the probability	73
A.2	Protocols with reverse reconciliation	74
A.2.1	Calculation of the probability with Mathematica and of the fidelity between Alice and Bob	74
A.2.2	Analysis of the different protocols	86

Introduction

Since the antiquity, the need of privacy and confidentiality for communications has been constantly increasing and nowadays, cryptography is part of everyone's life. Such an important topic motivated all the researches that were done on the subject. Many methods were invented in order to be able to share secret messages between two people. However, most methods have led to failures: a code that was secure at a certain time was often decrypted years after when the technologies had evolved. This implied an increasing complexity for the algorithms used. The first algorithms have been found in classical mechanics. However, the increasing complexity of the technology at the disposal of the eavesdropper may lead to failures of such protocols. In particular, when the quantum computers will be available for the eavesdropper, a big part of the current cryptography ciphers will collapse and not be secure anymore. This implies the need to find another source of security by using quantum key distribution protocols to share a secret key between two parties so that they can encode a confidential message. Quantum key distribution protocols involve quantum mechanics as well as information theory.

The difficulty is to obtain this secret key without seeing each other. Quantum mechanics offers the possibility to create such a key between two parties (Alice and Bob) without giving information to the eavesdropper (Eve) due to fundamental principles in quantum cryptography: the no-cloning theorem, the fact that any measurement will perturb the state... The first protocol that was invented was a protocol with discrete variables named BB84 due to the names of its inventors Bennett and Brassard and the year of its creation 1984. However, what was more developed during the next years is the use of continuous variables. The first protocol using continuous variables that was invented is the perfect analogue of BB84. It was developed at QuIC in 2000. Thereafter, others were developed but the work was never done to go back from these continuous variables protocols to their discrete analogues. This is precisely the objective of this thesis: come back to the discrete protocols and analyse them in order to determine if they could do better than BB84. Another objective is to study the reverse reconciliation. Indeed, for now only direct reconciliation was considered for discrete protocols.

The thesis is divided in two parts: in the first part we present the notions that will be used in this work as well as the state of the art in quantum key distribution while the second part is made of original contributions and the results found during this year. In the first part, the two first chapters will focus on an overview of all the basic concepts of quantum mechanics and of the theory of information which will be used: the first chapter presents the basics of quantum mechanics such as what is a qubit, what kind of measurements can be done,... The second one is about the theory of information of Shannon. In chapter 3, we make a review of quantum key distribution. We start by presenting in more details its objective. Then, we present BB84 and the protocols with continuous variables that were invented. Finally, in the last chapter of the first part, we speak about quantum cloning, as the action of the eavesdropper will be represented by a quantum cloning machine. The second part contains two important chapters as well as a discussion and a conclusion. In the first chapter, we search for the analogue

of the protocols for continuous variables. This allows us to define the protocols that will be studied in this thesis. Thereafter, the security is studied in the case of direct reconciliation. The second chapter is entirely dedicated to the reverse reconciliation.

Part I

Background and state of the art for quantum key distribution

Chapter 1

Basics of quantum mechanics

In this chapter, we present all the key notions in quantum mechanics that will be used in the next part of the thesis while presenting the various quantum cryptography protocols. The concepts used in this work are presented from the simplest one (the definition of a qubit) to the more complex properties of quantum mechanics such as the different methods of measurement that exist.

1.1 Definition of a qubit

In classical computation, the information is coded in terms of bits which are binary numbers: they can be equal to 0 or to 1. In quantum mechanics, we can find the equivalent of the bit: the quantum bit (more often written qubit). Here, we will do a mathematical description of the qubit. However, a qubit does correspond to a physical system as for example the spin of an electron. Like the classical bit which is in a "state" (0 or 1), the physical system will be in a state as well. The two basic states of a qubit are $|0\rangle$ and $|1\rangle$ which are written in the Dirac formalism. The qubit is then a two-level quantum system associated with a two-dimensional Hilbert space. However, unlike in classical mechanics, the qubit can be in any superposition of these two states:

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \quad \text{where } \alpha, \beta \in \mathbb{C} \quad (1.1)$$

The states $|0\rangle$ and $|1\rangle$ thus form a basis which is called the computational basis.

When measuring the qubit, the result will be the state $|0\rangle$ with a probability $|\alpha|^2$ and the state $|1\rangle$ with a probability $|\beta|^2$. As the sum of the probabilities must be equal to one, there is a condition on the values of α and β :

$$|\alpha|^2 + |\beta|^2 = 1 \quad (1.2)$$

An important representation of the qubit state is the Bloch sphere which is a geometrical approach. Indeed, as we have the condition 1.2, it is possible to rewrite the superposition state of the qubit as:

$$|\psi\rangle = e^{i\phi} \left(\cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) e^{i\gamma} |1\rangle \right) \quad (1.3)$$

In quantum mechanics, it is well known that the global phase is irrelevant as the state $|\phi\rangle$ is totally indistinguishable from the state $e^{i\theta} |\phi\rangle$ by any physical procedure. This means that the global phase will not change anything and that our state is equal to:

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) e^{i\gamma} |1\rangle \quad (1.4)$$

This equation corresponds to the equation of a point on a sphere: the Bloch sphere which is represented on figure 1.1.

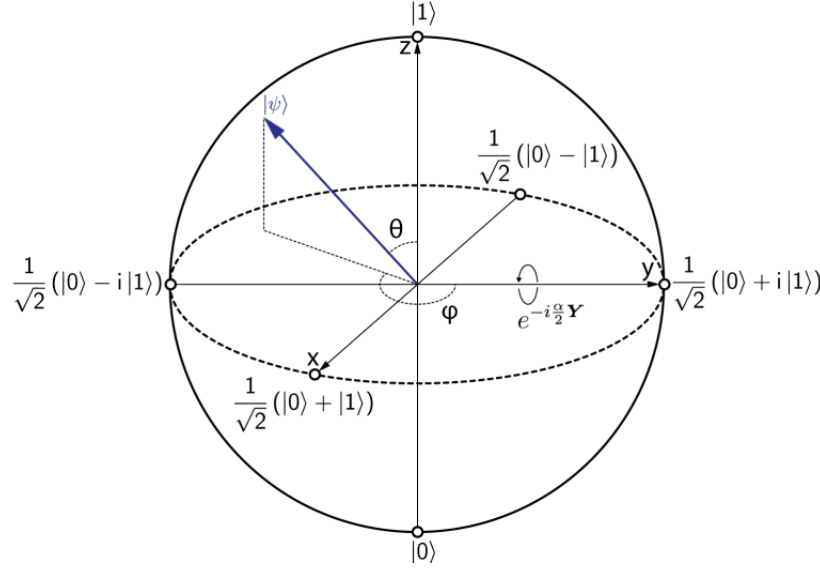


Figure 1.1 – Representation of all the possible states for a qubit on the Bloch sphere.

In quantum cryptography, qubits are at the heart of the protocols with discrete variables as they are the physical systems which are sent by Alice to Bob where Alice and Bob are our two correspondents. In the first protocol described in chapter 3, the computational basis is used as well as the dual basis which is composed of the states $|+\rangle$ and $|-\rangle$ which are:

$$\begin{cases} |+\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \\ |-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}} \end{cases} \quad (1.5)$$

In this thesis, another notation will also be used to define the state of a qubit: the vectorial form. Indeed, each ket can be expressed in the form of a vector. If:

$$\begin{cases} |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{cases} \quad (1.6)$$

then the state $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ is equal to:

$$|\psi\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (1.7)$$

1.2 Density matrix

Until now, we have represented the states of a system by a state vector $|\psi\rangle$ but there exists another way of presenting a state: the density matrix. This representation is more general than the vector state. Indeed, if the system is represented by the vector $|\psi\rangle$; the system is in a pure state as one vector is sufficient to describe it. However, in more general cases, the system is in a statistical mixture of several quantum states: $|\psi_1\rangle, |\psi_2\rangle, \dots$ and the probability to be in one of these quantum states is given by p_i where $i=1,2,\dots$

The representation of the system with a density matrix is valid even if it is in a statistical mixture: the density matrix is an operator which can characterize any quantum state. It is defined as:

$$\hat{\rho} = \sum_i p_i |\psi_i\rangle \langle \psi_i| = \sum_i p_i \hat{P}_i \quad (1.8)$$

where \hat{P}_i is the projector on the state $|\psi_i\rangle$. For pure states, the density operator is just equal to the projector on the state and there is only one probability p_1 which is equal to one.

The density operator has three important properties that are well known:

1. $\hat{\rho}$ is an hermitian operator: $\hat{\rho} = \hat{\rho}^\dagger$. This implies that the eigenvalues of the operator are real.
2. $\text{Tr}(\hat{\rho}) = 1$ due to the normalization of the operator. The sum of the eigenvalues is then equal to 1. 'Tr' is the trace of the operator and it will be defined in another section.
3. The last property is that the operator is positive: $\hat{\rho} \geq 0$. It is also valid for the eigenvalues: $\lambda_i \geq 0$.

The density operator formalism is useful in this thesis for the calculation of the proofs of security of different protocols.

1.3 Composite systems and quantum entanglement

Often in this thesis, quantum states composed of more than one system will appear as each person involved in the protocol will have his own qubit. This explains the importance of seeing how such systems are described. Moreover, when dealing with more than one qubit, the entanglement appears. This is a kind of correlation between the two states: the measurements on the two states are correlated even if the two qubits are separated by a very long distance. Entanglement is a phenomenon that cannot be found in classical mechanics. Therefore, it is used to obtain results that do not exist in classical physics. The first part of this section is dedicated to the presentation of some basics of a two qubits system as well as a few words about the entanglement. In the last part, we define the tensor product as it is used to obtain the vector or the density matrix of a system containing more than one qubit.

1.3.1 Bases for two qubits and entanglement

The traditional basis that we can define is the computational basis for a system of two qubits: $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ but this is only one of the many bases which exist.

Another important basis is the one composed by the Bell states:

$$\begin{cases} |\Phi_+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}} \\ |\Phi_-\rangle = \frac{|00\rangle - |11\rangle}{\sqrt{2}} \\ |\Psi_+\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}} \\ |\Psi_-\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}} \end{cases} \quad (1.9)$$

These states are very important in quantum mechanics as they are a perfect example of "maximally entangled" states.

This can be seen with the first state of the Bell basis: $|\Phi_+\rangle$: we imagine that one of the qubit is given to Alice and the second to Bob. Then, they go to two different places which are far from each other. If Alice does a measurement of her qubit in the computational basis, she has two possible outcomes: $|0\rangle$ or $|1\rangle$ each with a probability of one half. Immediately after the measurement of Alice, Bob can perform the same measurement on his own qubit. Then, he will always obtain the same result as Alice. Here, we took the example of a measurement in the computational basis but the result will be the same for any measurement: Alice can always know the state that Bob has from the result of her own measurement. This phenomenon is impossible to reproduce in a classical framework.

In a more general way, an entangled system is a system that is not separable and a system is separable if it can be put as a tensor product of two states: $|\phi\rangle = |\phi_1\rangle \otimes |\phi_2\rangle$. One important application of the entanglement which is worth mentioning here is the quantum teleportation. Quantum teleportation is a mean of transferring an unknown quantum state from Alice to Bob without modifying it. In order to do that, Alice and Bob just need to share an entangled state and to send each other classical bits. Quantum teleportation has already been successfully demonstrated experimentally.

1.3.2 Tensor product

If two states are under the form of a vector state:

$$|\Psi_1\rangle = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}; |\Psi_2\rangle = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \quad (1.10)$$

it is possible to find the vector for the whole system by applying a tensor product:

$$|\Psi_{tot}\rangle = |\Psi_1\rangle \otimes |\Psi_2\rangle = \begin{pmatrix} a_1 b_1 \\ a_1 b_2 \\ a_2 b_1 \\ a_2 b_2 \end{pmatrix}. \quad (1.11)$$

This definition of a tensor product can be generalized to any size of the vectors:

$$\vec{a} \otimes \vec{b} = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \otimes \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} = \begin{pmatrix} a_1 b_1 \\ \vdots \\ a_1 b_m \\ a_2 b_1 \\ \vdots \\ a_n b_m \end{pmatrix}. \quad (1.12)$$

An important property of the tensor product is that it is non-commutative. It is therefore necessary to be cautious when applying it to be sure to do it in the right way.

If the two systems are described by a density matrix, the density matrix of the whole environment can also be found with the use of a tensor product if there is no correlation

between the two systems:

$$\rho_{\hat{A}B} = \hat{\rho}_A \otimes \hat{\rho}_B \Leftrightarrow \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \otimes \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{s1} & b_{s2} & \cdots & b_{sp} \end{pmatrix} = \begin{pmatrix} a_{11}b_{11} & a_{11}b_{12} & \cdots & a_{11}b_{1p} & a_{12}b_{11} & \cdots & a_{1n}b_{1p} \\ a_{11}b_{21} & a_{11}b_{22} & \cdots & a_{11}b_{2p} & a_{12}b_{21} & \cdots & a_{1n}b_{2p} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{m1}b_{s1} & a_{m1}b_{s2} & \cdots & a_{m1}b_{sp} & a_{m2}b_{s1} & \cdots & a_{mn}b_{qp} \end{pmatrix} \quad (1.13)$$

This is really useful when treating problems with more than one qubit as it allows us to calculate the state of the total system.

1.4 Trace and partial trace

Another important tool that will be used in this thesis to demonstrate the limits of the security of a protocol of communication is the partial trace. Before defining this concept, it is important to remind the definition of a simple trace:

The trace of an operator corresponds to the trace of the matrix of this operator. If \hat{A} is an operator and $|n\rangle$ is any orthonormal basis, the trace of the operator is simply:

$$\text{Tr}(\hat{A}) = \sum_n \langle n | \hat{A} | n \rangle \quad (1.14)$$

This definition holds for any basis as it can be demonstrated by using the closing relationship.

Starting from this, it is now possible to define the partial trace of a system. This mathematical concept is very strong as it makes it possible to describe any subsystems of a big system if we have the description of the big one. It means that if we have the density matrix for the total system, it is possible to calculate the one of any subsystem.

For example, if A and B are two subsystems of a bigger quantum system AB. We obtain the density operator of A by tracing out B with a partial trace. This means:

$$\hat{\rho}_A = \text{Tr}_B(\rho_{\hat{A}B}) = \sum_{i^B} \langle i^B | \hat{\rho}_{AB} | i^B \rangle \quad (1.15)$$

where $|i^B\rangle$ is an orthonormal basis of the Hilbert space of B.

The partial trace exhibits some important properties:

- It is independent of the order of the applications of the different partial traces:
 $\hat{\rho}_C = \text{Tr}_B(\text{Tr}_A(\hat{\rho}_{ABC})) = \text{Tr}_A(\text{Tr}_B(\hat{\rho}_{ABC}))$
- If we make the partial trace on all the subsystems, we obtain the total trace:
 $\text{Tr}(\rho_{\hat{A}B}) = \text{Tr}_B(\text{Tr}_A(\hat{\rho}_{AB}))$

1.5 Measurements

In many protocols in quantum communication as well as in all the algorithms in quantum computation, the last step is the same: a measurement. This is very important as it

is the step that will give us information under the classical form: bits. It makes the link between quantum states and the classical world. For example, in quantum key-distribution protocols, the result of the measurement is a bit string that will then be used to encode messages as explained in section 3. This justifies the need to detail what is a measurement in quantum cryptography.

In quantum mechanics, one of the fundamental postulates involves the measurement. This postulate is formulated in [27] as:

"Quantum measurements are described by a collection M_m of measurement operators. These are operators acting on the state space of the system being measured. The index m refers to the measurement outcomes that may occur in the experiment. If the state of the quantum system is $|\Psi\rangle$ immediately before the measurement then the probability that result m occurs is given by

$$p(m) = \langle \Psi | M_m^\dagger M_m | \Psi \rangle \quad (1.16)$$

and the state of the system after the measurement is

$$\frac{M_m |\Psi\rangle}{\sqrt{\langle \Psi | M_m^\dagger M_m | \Psi \rangle}}. \quad (1.17)$$

The measurement operators must fulfil the completeness relation:

$$\sum_m M_m^\dagger M_m = \mathbb{1} \quad (1.18)$$

Here, everything has been expressed for a state vector but the postulate can be formulated in terms of the density operator of the state. Indeed, if the state is described by ρ just before the measurement, the probability of measuring m is given by:

$$p(m) = \text{Tr}(\rho M_m^\dagger M_m) \quad (1.19)$$

and the state after the measurement is:

$$\rho_m = \frac{M_m \rho M_m^\dagger}{\text{Tr}(M_m^\dagger M_m \rho)} \quad (1.20)$$

In this thesis, this formulation in terms of density matrix will be used.

Depending on the operators M_m , two main types of measurements can be distinguished: the projective measurement and the positive-operator valued measurement. They will be detailed in the two next subsections.

1.5.1 Projective measurement

Projective measurements are a special case of the postulate presented here above. However, most people think about these projective measurements when speaking of measure as it is the best known measurement method. These methods are also called "Von Neumann measurement" or "positive-valued measurement". In this kind of measurement, the observable that is used is:

$$O = \sum_m m P_m \quad (1.21)$$

where P_m is a projector on an eigenstate of O with eigenvalue m . P_m has two important properties: it is hermitian ($P_m^\dagger = P_m$) and idempotent ($P_m^2 = P_m$) which is due to the fact that this is a projector.

The result of the measurement is m with a certain probability:

$$p(m) = \langle \Psi | P_m^\dagger P_m | \Psi \rangle = \langle \Psi | P_m | \Psi \rangle. \quad (1.22)$$

The state after such a measurement that gave m as a result is:

$$|\Psi_m\rangle = \frac{P_m |\Psi\rangle}{p(m)} = \frac{P_m |\Psi\rangle}{\langle \Psi | P_m | \Psi \rangle} \quad (1.23)$$

A thing that can be mentioned is that such a measurement is reproducible. Indeed, if we do the same measurement again just after the first one, the result will be exactly the same.

We can see an advantage of such measurements: Von Neumann measurements allow us to distinguish perfectly orthogonal states such as for example $|0\rangle$ and $|1\rangle$. However, this leads to the main defect of these measurements: they cannot be used to distinguish states that are not orthogonal.

This can be understood by observing a simple example which is taken from the book [27]. Alice can prepare two states:

$$\begin{cases} |\Psi_1\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \\ |\Psi_2\rangle = |0\rangle \end{cases} \quad (1.24)$$

Then, she sends her qubit to Bob who wants to know what qubit he received. In order to achieve that objective, he will make a measurement with the projectors $P_0 = |0\rangle\langle 0|$ and $P_1 = |1\rangle\langle 1|$. The result will depend on the state that he received.

- He received $|\Psi_1\rangle$: He then has a probability of $\frac{1}{2}$ to obtain a 0 as the result and the same probability to obtain 1.
- He received $|\Psi_2\rangle$: There is only one possible result: 0.

We see that if Bob obtains a 0 by doing his measurement, it is impossible for him to know which state he received. Von Neumann measurement cannot be used to discriminate states that are not orthogonal. This flaw can be avoided by using a POVM instead of a Von Neumann measurement.

1.5.2 POVM (positive-operator valued measure) measurement

This kind of measurement is more general than the Von Neumann one. Unlike it, the POVM is not reproducible. We define the POVM elements

$$E_m = M_m^\dagger M_m \quad (1.25)$$

which form a set of POVM elements: $\{E_m\}$. They satisfy some properties:

- They are positive.
- Their sum gives the identity: $\sum_m E_m = \mathbb{1}$

The important property of these measurements is that it can be used to distinguish states that are non-orthogonal which was impossible for the projective measurement. We can show it using the same little example as before.

Alice sends one of these states to Bob:

$$\begin{cases} |\Psi_1\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \\ |\Psi_2\rangle = |0\rangle \end{cases} \quad (1.26)$$

Bob will not use projective elements this time but the elements of a POVM:

$$\begin{cases} E_1 = \frac{\sqrt{2}}{1+\sqrt{2}} |1\rangle \langle 1| \\ E_2 = \frac{\sqrt{2}}{1+\sqrt{2}} \frac{(|0\rangle-|1\rangle)(\langle 0|-\langle 1|)}{2} \\ E_3 = I - E_1 - E_2 \end{cases} \quad (1.27)$$

It is easy to show that $\sum_i E_i = \mathbb{1}$ and that these elements are positive as their eigenvalues are positive. Again, the result of the measurement will depend on the state that Bob received:

- He received $|\Psi_1\rangle$: Bob has no chance to measure E_2 as this element has been chosen such that: $\langle \Psi_1 | E_2 | \Psi_1 \rangle = 0$.
- He receives $|\Psi_2\rangle$: Following the same idea, Bob has no chance to measure E_1 .

If Bob obtains E_1 , he is sure that he received $|\Psi_2\rangle$ and inversely, if he measures E_2 , he must have received $|\Psi_1\rangle$. However, if the outcome is E_3 , he cannot interpret the result and find the state that he received.

Nevertheless, Bob will never make mistakes while trying to find what Alice sent him; there will just be some cases where he will not receive any information and this absence of errors will be very important.

1.6 Conclusion

In this chapter, we have presented all the fundamental notions of quantum mechanics that will be used in this thesis in order to study the protocols of quantum key-distribution protocols with discrete variables. We started with the notion of a qubit which is at the heart of all the protocols of quantum cryptography with discrete variables. Moreover, such protocols involve more than one qubit (each person involved has one qubit: Alice, Bob but also Eve; the eavesdropper). This justifies the fact that we have defined the systems with more than one variable. Finally, the concepts of density matrix, partial trace and measurements have been explained as they will be useful for the security proofs.

Chapter 2

Basics of information theory

We owe the theory of information to Shannon, who has published in 1948 an article called "The Mathematical Theory of Communication" [30]. This article is the origin of an entire mathematical theory around the concept of information. This theory is used here in order to interpret probability distributions by assigning them a physical value with an interpretation: the entropy of Shannon. This quantity is the main element of the whole theory and is useful when studying the security of protocols in quantum cryptography. The theory of information of Shannon is very wide but only a little part (the interesting one for this thesis) will be presented in this chapter. For example, the theory of information is valuable for discrete and continuous variables; however, only the case of discrete variables will be presented here.

2.1 Shannon entropy

Shannon entropy is the central quantity of this theory. It is used to measure the uncertainty on a variable. Let's imagine we have a discrete random variable X which can take the values x with a probability $p(x)$. Then, the entropy of this variable will be:

$$H(X) = - \sum_x p(x) \log_2(p(x)) \quad (2.1)$$

As we used a logarithm in base 2, this means that the entropy is expressed in term of bits. The entropy $H(X)$ is thus the number of bits needed to describe the variable.

A simple example often used is the coin tossing. If we toss a coin, there are two possible outputs, each with a probability of $\frac{1}{2}$. The entropy of the distribution is simply equal to:

$$H(\text{Coin tossing}) = - \sum_x p(x) \log_2(p(x)) = -\frac{1}{2} \log_2\left(\frac{1}{2}\right) - \frac{1}{2} \log_2\left(\frac{1}{2}\right) = 1 \quad (2.2)$$

To describe this variable, we just need one bit. We can for example associate a '0' with a head and a '1' to a tail.

For the definition of entropy to be complete, there is a convention that must be added. Indeed, when adding a term with a zero probability, the entropy must not change. This leads to the convention:

$$0 \log_2(0) = 0$$

This convention can also be explained by looking at the limit when x tends to 0 of $x \log_2(x)$:

$$\lim_{x \rightarrow 0} x \log_2(x) = 0 \quad (2.3)$$

The most important property of the entropy is that it is always positive: $H(X) \geq 0$. Indeed, as $0 \leq p(x) \leq 1$, $-\log p(x) \geq 0$. This can be seen on the graph 2.1.

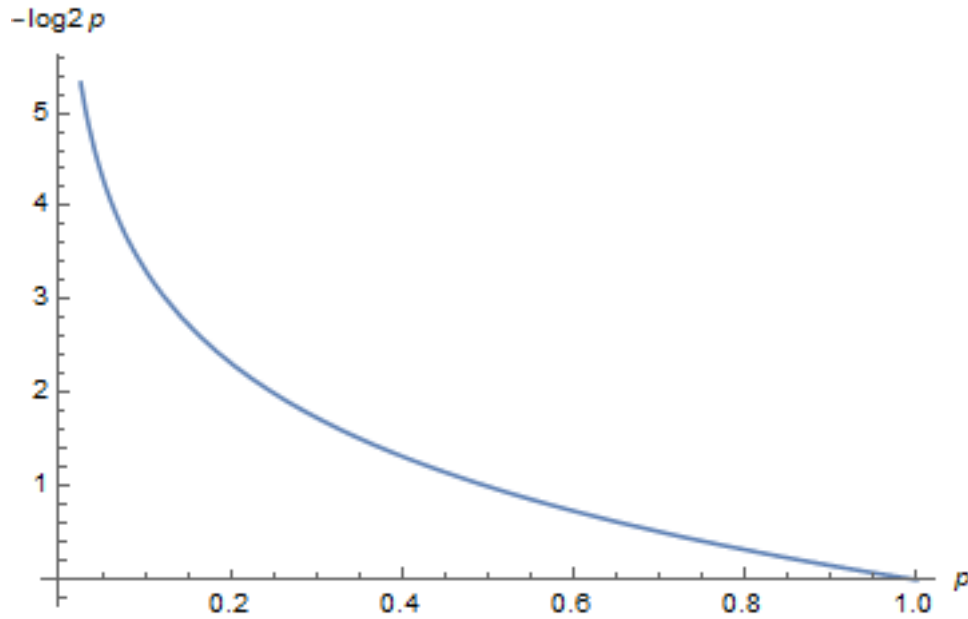


Figure 2.1 – Values of $-\log_2(p)$ as a function of the value of p .

2.2 Joint and conditional entropy

The definition given in the previous section can be extended to the case of more than one variable. This leads to the definition of joint entropy. If X and Y are two discrete variables, the joint entropy is defined as:

$$H(X, Y) = - \sum_x \sum_y p(x, y) \log_2(p(x, y)) \quad (2.4)$$

where $p(x, y)$ is the joint probability distribution. As for the entropy of a single variable, $H(X, Y)$ represents the uncertainty on the pair of variables (X, Y) .

The definition of joint entropy has been introduced here in the case of two variables, as it will be used later in this work, but it can be rewritten for an arbitrary number of random variables:

$$H(X_1, X_2, \dots, X_n) = - \sum_{x_1} \sum_{x_2} \dots \sum_{x_n} p(x_1, x_2, \dots, x_n) \log_2(p(x_1, x_2, \dots, x_n)) \quad (2.5)$$

Another important quantity is the conditional entropy of a discrete random variable Y with respect to another discrete random variable X . Its definition is:

$$H(Y|X) = \sum_x p(x) H(Y|X = x) \quad (2.6)$$

This expression can be rewritten if we take into account the well-known property of a conditional probability distribution:

$$p(x, y) = p(x) \cdot p(y|x) \quad (2.7)$$

Thanks to that, we rewrite:

$$\begin{aligned}
H(Y|X) &= \sum_x p(x) H(Y|X=x) \\
&= \sum_x p(x) \left(- \sum_y p(y|x) \log_2(p(y|x)) \right) \\
&= - \sum_x \sum_y p(x,y) \log_2(p(y|x))
\end{aligned} \tag{2.8}$$

This conditional entropy indicates the uncertainty on Y if we know X . This entropy is not equal to $H(Y)$ because some information on Y can be contained in X .

There exists a link between the conditional and the joint entropies:

$$H(X, Y) = H(X) + H(Y|X) \tag{2.9}$$

This results from the relation 2.7 and the relation $\sum_y p(x,y) = p(x)$. Consequently, the relation 2.9 can be demonstrated by starting with the definition of the joint entropy:

$$\begin{aligned}
H(X, Y) &= - \sum_x \sum_y p(x,y) \log_2(p(x,y)) \\
&= - \sum_x \sum_y p(x,y) \log_2(p(x) \cdot p(y|x)) \\
&= - \sum_x \sum_y p(x,y) \log_2(p(x)) - \sum_x \sum_y p(x,y) \log_2(p(y|x)) \\
&= - \sum_x p(x) \log_2(p(x)) + H(Y|X) \\
&= H(X) + H(Y|X)
\end{aligned} \tag{2.10}$$

This proves the relation 2.9 that will be used in the next section.

2.3 Mutual information and its interpretation

One last quantity of the Shannon information theory that will be used is the mutual information between two discrete variables. This is defined as:

$$I(X : Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \tag{2.11}$$

Using the relation 2.9, it is possible to rewrite the mutual information as:

$$I(X : Y) = H(X) + H(Y) - H(X, Y) \tag{2.12}$$

The mutual information represents the information shared between the two variables X and Y . Making the link between this and the definition 2.12 is quite easy. Indeed, when adding up $H(X)$ and $H(Y)$, we obtain the total uncertainty on X and on Y . The part of the information common to X and Y is thus added twice. By removing the joint entropy, we remove the own uncertainty of X and Y and once their shared information. As a result, the common information between X and Y is kept only once. This is called the mutual information.

All the quantities that were defined before can be represented on a Venn diagram, as represented on figure 2.2.

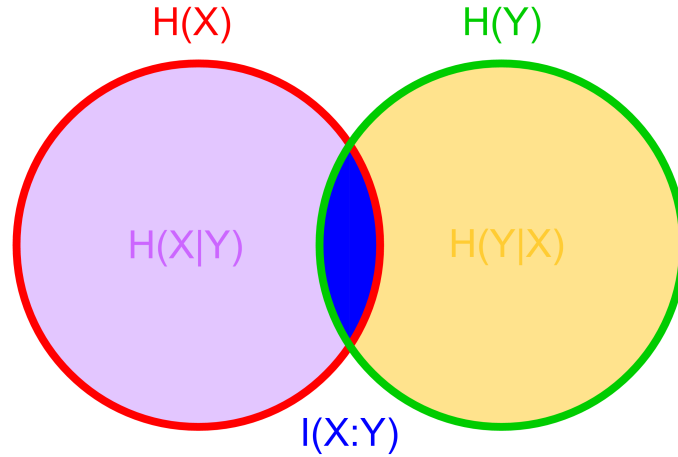


Figure 2.2 – Venn diagram of the different entropies.

On this diagram, the different relations that were shown before appear clearly. Indeed, the relation 2.9 is then equivalent to the sum of the two circles; the common part of them is summed two times. When we remove the joint entropy which is the union of the two circles, the surface which is common will be removed only once. The remaining surface is then equal to the mutual information.

2.4 Csiszar-Körner theorem for the distillation of a secret key

The Csiszar-Körner theorem has been proposed by Csiszar and Körner in an article in 1978 [11]. It gives a sufficient condition for a protocol of cryptography to work. The theorem is explained in this chapter because it is entirely based on the mutual information.

To understand it fully, identifying which parties are involved in the protocol is required. First of all, there are two authorized correspondents: Alice and Bob. Their objective is to send a secret key from Alice to Bob as will be explained more in details in chapter 3. The last party is Eve; she is an eavesdropper who wants to intercept the message.

According to the protocol used and the method used by Eve to spy the communication, Alice will share some amount of information with Bob and some with Eve. These amounts are described by mutual information: I_{AB} between Alice and Bob and I_{AE} between Alice and Eve.

The Csiszar-Körner theorem says that a secret communication is possible only if:

$$I_{AB} - I_{AE} > 0 \quad (2.13)$$

If this condition is fulfilled, it is always possible to send a message from Alice to Bob by doing a post-treatment on the bits they have. Such a condition seems to be logical as it simply means that Alice must share more information with Bob than with Eve. However, if this condition is not fulfilled, this does not mean that it is impossible to send a secret key between Alice and Bob. Indeed, in some particular cases, it is still possible to send a secret key but at a much lower bit rate. In this work, the protocols will be studied in order to see if the condition is fulfilled and thus to see if it is possible to extract a secret key for sure.

Chapter 3

Quantum cryptography protocols

3.1 Introduction to quantum cryptography: origins and objective

In today's world, cryptography takes a significant place. Everyday, everyone is confronted to cryptography even without knowing it. For example, each time someone does a payment with his card or on-line, he uses cryptography,... This reveals the importance of cryptography in our society nowadays. The objective of cryptography is to provide privacy, confidentiality and authentication by coding messages. This need exists since thousands of years as it is proved by the first example known of cryptography: the Caesar code that comes from the antiquity.

Today, there exist two kinds of protocols in classical cryptography: the symmetric ones and the antisymmetric ones.

- **Symmetric protocol:** In this class, the message that is sent is coded by Alice using a secret key and is decoded by using the same key. The difficulty of such protocols is to obtain the secret key. It cannot be known by an eavesdropper otherwise he could intercept the message. An important example of this kind of protocols is the Vernam cipher [32]: Alice and Bob must possess a key composed of random bits that must be as long as the message that they want to encode. This key can only be used once to encode a message. For the next message, they must then have another key. Alice adds this key to the message and Bob must extract the same key from the encoded message. The flaw of this method is the need to share the key between Alice and Bob by a secure method. However, it has been proved to be unconditionally secure if the conditions on the keys are respected (it is as long as the message, totally random and used only once). "Unconditionally secure" means that it is secure whatever power the eavesdropper has.
- **Asymmetric protocols:** Such protocols involve two keys: a secret key to decode the message and a public one to encode it. The person that must receive the message will generate the two keys. She will keep the secret key and transmit the public one to the person that must encode the message. Once the message is encoded, it is very hard to decode it unless we have the secret key. Unlike in symmetric cryptography, there is no difficulty to share the key. One of the most famous ways to do asymmetric cryptography is the RSA algorithm named after its inventors: Rivest, Shamir and Adleman [28]. The security of this protocol is entirely based on the fact that factorizing a huge number is extremely complicated for a modern computer. By the time a computer could have factorized this number, the message would be useless for the spy.

As all these classical protocols exist, one may ask: "Why do we need quantum cryptography?". This need is justified by the fact that the asymmetric protocols are not unconditionally secure. Indeed, if someday someone finds an algorithm allowing people to factorize any huge number on an efficient way, the security of the RSA algorithm will collapse. This is the same for all the asymmetric protocols. The problem is that an efficient algorithm has already been found for factorising numbers. This algorithm is the quantum Shor algorithm [31]. This means that when quantum computers will become available, all the cipherings used nowadays will collapse and they will become useless. There are two alternatives to this absence of security: either the researchers find classical protocols of communication that are unconditionally secure; either we use quantum cryptography protocols. This second option has been in development since the 80s and the researches have led to some impressive results and even practical applications as for example in [13] or in [25]. This thesis will be focused on quantum cryptography. A good review of the basics of quantum cryptography can be found in [34].

3.2 Basic principle

"Quantum cryptography" is not the most accurate word to describe the protocols about which we talk. Indeed, it is more general than the subject of this work where only the quantum key-distribution protocols will be studied. These protocols do not allow Alice and Bob to share a message between them but only a secret key.

This means that using such protocol Alice and Bob will share a secret key (and not a message) that they will then use to apply the Vernam method (which is unconditionally secure). Two categories of protocols can be distinguished: the protocols using discrete variables and the ones using continuous variables. For discrete protocols, qubits are used. Physically, this discrete variable can correspond to the spin of an electron or more often to the polarization of a single photon. For continuous protocols, the information can be for example stored in the quadratures of light. The first methods that were invented are the discrete ones. However, the ones with continuous variables are easier to use as it is easier to produce a coherent light with a laser than a single photon.

Whatever the kind of method used, the general method is always the same: Alice and Bob have to share two channels, a classical one and a quantum one as represented on the figure 3.1. The classical line must be secure: Eve (the spy) can listen to what is sent on this line but cannot interact with what is on it. This is why Alice and Bob must authenticate themselves by sending a message on the line. On the other hand, Eve can interact with the quantum line and she is only limited by the laws of physics. An advantage in quantum mechanics is that any interaction of Eve will affect the state so that Alice and Bob will be able to discover her presence. Indeed, if Eve extracts information about the state this can be seen as a general measurement and we clearly saw in chapter 1 that a measure modifies the state. As a result, Bob won't receive the same state as the one sent by Alice, and they will be able to detect it, as the results of the measurements of Bob won't always fit with what was expected by Alice, due to the action of Eve. Someone could then say that instead of extracting information from the state, Eve could just copy the state and make her measurement on her own. This is impossible due to a fundamental theorem in quantum mechanics: the no-cloning theorem. This theorem will be explained in chapter 4.

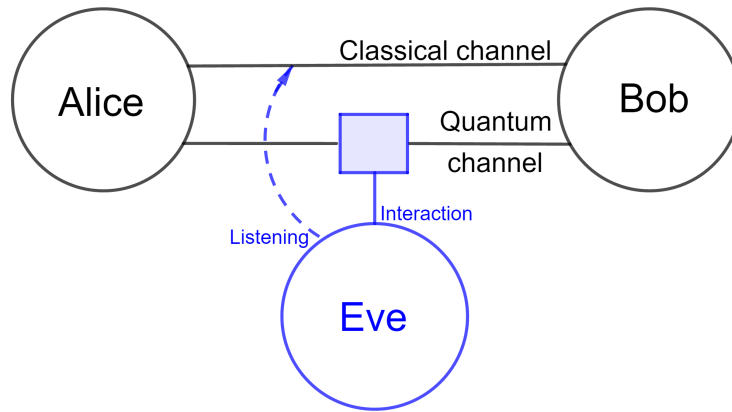


Figure 3.1 – Representation of a classical configuration.

Usually, a protocol is done in 6 steps:

- **Preparation of the state by Alice:** Alice prepares the state that she will send to Bob. This often involves the use of random numbers to choose the basis and the states that she sends. This stage can be seen in a totally different way, that is exactly equivalent, which involves that Alice has a generator of entangled states and that she measures one of the states generated and sends the other to Bob.
- **Transmission of the state on the quantum channel:** Alice sends the state on the line and Eve can interact with the state.
- **Measure of Bob:** Bob measures the state. This can imply a random number if Bob must choose the basis in which he will make the measurement. At this stage, Alice and Bob both have a bit string which are more or less correlated in function of the parameters of the line and of what Eve did.
- **Exchange of information to remove some bits:** Alice and Bob share some information about their measurement. For example, in the first scheme that we will see (BB84), they share the basis that they used and they keep the corresponding bit only if they used the same basis. At this stage, Bob obtains a key called "sifted key".
- **Disclosure of certain bits on both sides:** Alice and Bob will reveal some bits of their key (the same ones). This will allow them to characterize the line and estimate the bit error rate (QBER). The QBER is the probability to have an error (thus different bits between Alice and Bob). They will also be able to discover the presence of Eve. An hypothesis that is always done is that the situation is the worst possible: all the errors are due to Eve and not to the line. They can then use the theorem of Csiszar-Körner (equation 2.13) to determine if they will be able to create a secret key or not.
- **Classical post processing:** Thereafter, Alice and Bob use the classical channel to obtain a common key. To do that, they use algorithms for error correction and privacy amplification. There are two kinds of reconciliation which are possible even if only one has been studied before for discrete variables: the direct reconciliation and the reverse reconciliation. In the direct reconciliation method, the final key is built from the one possessed by Alice and in the reverse reconciliation it is built

from the bits possessed by Bob. After this classical step, Alice and Bob obtain the final key.

One of the objective of this thesis is to study the reverse reconciliation in the case of the discrete variables, in order to observe if it allows us to obtain a better communication; a better bit rate. The bit rate between Alice and Bob is defined in the case of the direct reconciliation case by:

$$\text{Bit rate} = \max(I_{AB} - I_{AE}, 0) \quad (3.1)$$

Analogously, for the reverse reconciliation, the bit rate is:

$$\text{Bit rate} = \max(I_{BA} - I_{BE}, 0) \quad (3.2)$$

If the bit rate is equal to 0, this means that they cannot extract a secret key according to the Csiszar-Körner theorem and that they need to stop the communication. This bit rate equal to zero is due to the fact that Eve has gained too much information.

In the next sections, we will briefly express the different actions that Eve can do to obtain information. When analysing a protocol, these possible attacks must be taken into account and we must demonstrate that the protocol is secure against all of them. Thereafter, we will briefly explain the most important protocols that have been invented with discrete and continuous variables.

3.3 Classification of possible attacks

There exist three kinds of attacks that Eve can perform:

- Individual attack: Eve interacts independently with each state sent by Alice, makes a measurement on her probe, and then sends the system to Bob. The main restriction is that Eve's measurement is done before Alice and Bob communicate on the classical channel.
- Collective attack: Again, she interacts independently with each state but now she has a quantum memory. She can store the state that she obtained after an interaction and she can make the measurement on all the states collectively. This means that she can wait that Alice and Bob reveal some information on the classical channel before making the best measurement possible.
- Coherent attack: this is a more complex attack. Eve's probe system interacts on all the states sent by Alice and is then stored in a quantum memory. Eve later measures her probe only after the communication between Alice and Bob on the classical channel.

Studying the two last kinds of attack is more difficult. In this work, for the protocols that will be proposed, only the individual attacks will be analysed.

3.4 Discrete variables protocols

The first protocol proposed to do quantum cryptography involves discrete variables. It was proposed in 1984 by Bennett and Brassard [2] what has led to its name: BB84. This was the starting point of a lot of researches to establish protocols but also the fact that they are secure against any attack of Eve.

After BB84, other protocols with discrete variables were proposed but it still remains a reference in terms of protocols with discrete variables. This is why it will be presented in detail in the next subsection.

3.4.1 BB84

This protocol uses the computational basis $\{|0\rangle, |1\rangle\}$ and the dual basis $\{|-\rangle, |+\rangle\}$. In each of these bases, the bit '0' is assigned to a state and the bit '1' to the other. For example, $|0\rangle$ and $|-\rangle$ code a '0' and $|1\rangle$ and $|+\rangle$ code a '1'. The two bases are not orthogonal:

$$\begin{cases} \langle 0|+ \text{ or } -\rangle = \langle 0| \frac{|0\rangle \pm |1\rangle}{\sqrt{2}} = \frac{1}{\sqrt{2}} \\ \langle 1|+ \text{ or } -\rangle = \langle 1| \frac{|0\rangle \pm |1\rangle}{\sqrt{2}} = \pm \frac{1}{\sqrt{2}} \end{cases} \quad (3.3)$$

As these are non-orthogonal, it is impossible to distinguish them with a projective measurement.

We will first study the basic protocol in the absence of Eve by showing step by step what is done by Alice and Bob:

- **Preparation of the state:** Alice sends a qubit to Bob from one of the two bases. Alice chooses the state that she sends randomly.
- **Measure of Bob:** When Bob receives the state, he performs a projective measurement in one of the two bases. If Bob measures in the same basis as the one used by Alice, he is sure to obtain the same bit as her. If he chooses the bad basis, he has one chance out of two to obtain a bad result. This can be seen by calculating the probabilities with the equation 1.22 for an example. If Alice sends the state $|+\rangle$; she has a bit 1. Bob then measures in the bad basis (the computational one). He has a probability of $\frac{1}{2}$ to obtain the state $|0\rangle$ (he has then a bit 0) and the same probability to obtain the state $|1\rangle$ (he obtains a bit 1). Half of the time, the bit that he obtained is not the same as the one of Alice. The value $\frac{1}{2}$ was found with the formula 1.22:

$$\begin{cases} p(\text{Result} = |0\rangle \mid \text{Alice's state} = |+\rangle) = \langle + | (|0\rangle \langle 0|) | + \rangle = |\langle 0|+ \rangle|^2 = \frac{1}{2} \\ p(\text{Result} = |1\rangle \mid \text{Alice's state} = |+\rangle) = \langle + | (|1\rangle \langle 1|) | + \rangle = |\langle 1|+ \rangle|^2 = \frac{1}{2} \end{cases} \quad (3.4)$$

This value of $\frac{1}{2}$ is a known probability for mutually unbiased bases. Two bases are mutually unbiased if when we measure in one basis a state that was encoded in the other basis, we have the same probability to obtain each output.

- **Bases reconciliation:** The two first steps are repeated n times so that Alice and Bob have each a bit string with n bits. Alice and Bob then communicate via the classical channel. They compare the basis that they used for each state. If they used the same basis, they keep their bit. Else, they discard it. In average, they both end with a bit string of size $\frac{n}{2}$.

The other steps are not needed if there is no eavesdropper (but Alice and Bob could not know it for sure). This is why there is always another step: Alice will reveal some of her bits on an arbitrary way. Bob compares them to his bits. They can estimate the bit error rate (QBER) which will show them if they will be able to obtain a secret key. Indeed, the QBER is linked to the mutual information I_{AB} and I_{AE} which leads to the limit of security with the Csiszar-Körner theorem presented before. In the case without

eavesdropper, all the errors will be due to the defects of the line and will normally be very low.

After all these steps, there is also the classical post-processing with the error correction and the privacy amplification. Only the direct reconciliation has been studied with BB84: the key is produced from the bits owned by Alice.

Such a protocol is represented on the figure 3.3. To represent the different bases and the states, a convention is used that is presented on figure 3.2.

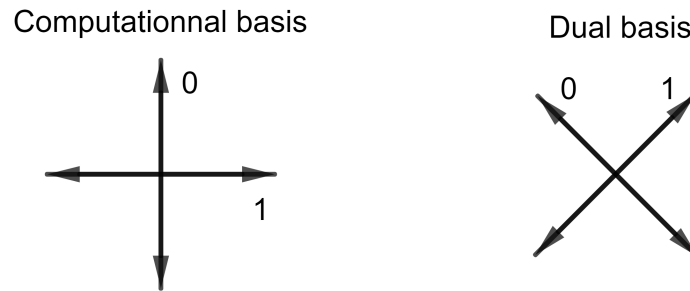


Figure 3.2 – Representation of the two bases used for the protocol. For each state, a bit '0' or '1' is associated .

ALICE						
Random bits	0	0	1	0	1	1
Random basis						
State sent						
BOB						
Random basis						
State measured						
Bit obtained	0	0	1	1	1	1

Figure 3.3 – Execution of BB84 protocol without Eve. In red, bits removed due to the bases reconciliation step.

In this example, it appears indeed that the bits obtained by Bob are exactly the same as the ones of Alice if they use the same basis.

We will now see an example where Eve is present and make an attack called "intercept and resend". It is an individual attack where Eve chooses randomly a basis. She measures

in this basis and then sends the state that she obtained to Bob. If she measures in the good basis; there is no difference with respect to what was obtained before: she sends to Bob exactly the same state that Alice has also sent. However if she measures in the wrong basis and Bob in the good one this can create errors because she sends to Bob a state in the wrong basis. He has one chance over two to obtain the wrong bit. By disclosing some of their bits Alice and Bob will thus be able to detect the presence of Eve. Such situation is represented on figure 3.4.

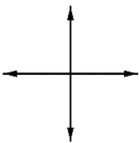

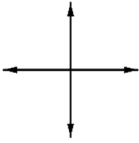
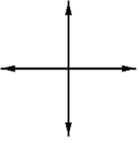

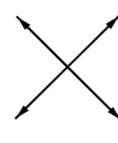






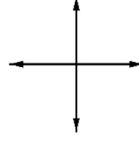


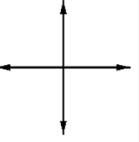

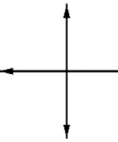


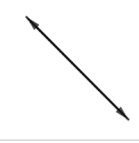



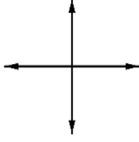

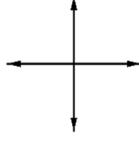
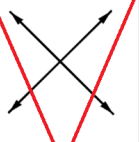
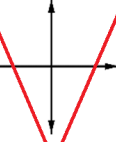
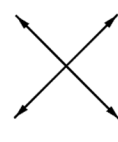



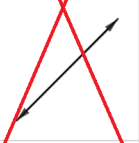
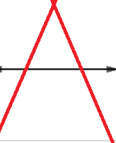

ALICE						
Random bits	0	0	1	0	1	1
Random basis						
State sent						
EVE						
Random basis						
State measured and sent to Bob						
BOB						
Random basis						
State measured						
Bit obtained	0	0	0	1	1	1

Figure 3.4 – Execution of BB84 protocol with Eve. In red, bits removed due to the bases reconciliation step. It appears that some bits obtained by Bob are different from the ones of Alice even if they used the same basis (in blue).

The bit in blue on the previous figure is different for Alice and Bob. They will detect it by disclosing some bits and this gives the QBER; the bit error rate.

However, in order to claim that the protocol is indeed secure, we need to connect that to the mutual information so that we can use the Csiszar-Körner theorem. This can be done due to the definition of a channel capacity. The channel capacity is defined as [10]:

$$C = \max_{p(\text{input})} I(A : B) \quad (3.5)$$

where $p(\text{input})$ is any input probability distribution which is possible. So the capacity is the maximal mutual information between the two correspondents that can be obtained by choosing the right input distribution of states. Here, the input probability distribution is fixed: Alice has one chance over four to send each state. The maximum disappears in the expression of the capacity and it is thus equal to the mutual information between Alice and Bob : $C = I(A : B)$.

If we know the capacity, we then have the mutual information. A well-known expression for the capacity exists when there is a probability p that the input is the same as the output. In this case, the capacity is given by:

$$C = 1 - h(p) \quad \text{where } h(p) = -p\log_2(p) - (1 - p)\log_2(1 - p). \quad (3.6)$$

This result is valuable for a binary symmetric channel. By injecting the definition of $h(p)$, this gives:

$$C = 1 + p\log_2(p) + (1 - p)\log_2(1 - p) \quad (3.7)$$

Here, p is equal to $1 - \text{QBER}$. Indeed, without eavesdropper, Bob obtains the same bit as Alice. However, when Eve is present, he obtains the same bit with a probability which is equal to one minus the QBER as the QBER gives the probability of errors. On figure 3.5, in blue, we have plotted the capacity of the line as a function of the QBER. We can also mention that the fidelity which is defined in chapter 4 is equal to "1-QBER" which will be useful in the numerical simulations of the second part of this thesis.

This proves that from an estimated QBER, Alice and Bob can know the mutual information between them by taking the ordinates of the curve for the estimated QBER.

To find the limiting QBER for the security of the protocol, I_{AE} is needed in order to apply the theorem of Csiszar-Körner. In many articles as in [18], the best individual attack that Eve can perform has been searched. In this same article, the method is to start from the fact that Eve applies any unitary operation on the state sent by Alice. The authors use then the symmetry of this unitary operation (as Eve wants to have the same effect on the two bases) to obtain the probability that Eve obtains the same result as Alice. This probability is equal to $\frac{1+\sin(x)}{2}$; where x is a real parameter describing the unitary operation done by Eve. Indeed, $\cos(x)$ is equal to the overlapping between the two states that Eve can obtain after the unitary operation for the two different input states received from Alice. By trying to maximise the mutual information between Alice and Eve, it is possible to find the expression of the QBER as a function of the same parameter.

$$\text{QBER} = \frac{1 - \cos(x)}{2}. \quad (3.8)$$

This method to find the values for the mutual information is quite difficult as at this time, all the unitary operations that could be done by Eve are considered and the objective is to optimize the result. A simpler way to model the action of Eve is now used in many articles: Eve just applies an optimal imperfect cloning to the state sent by Alice.

However, using again the formula 3.7, the complex method has led to an expression for I_{AE} which is equal to:

$$I_{AE} = 1 - h\left(\frac{1 + \sin(x)}{2}\right) \quad (3.9)$$

where x is found from the expression of the QBER. This mutual information has also been plotted on the figure 3.5.

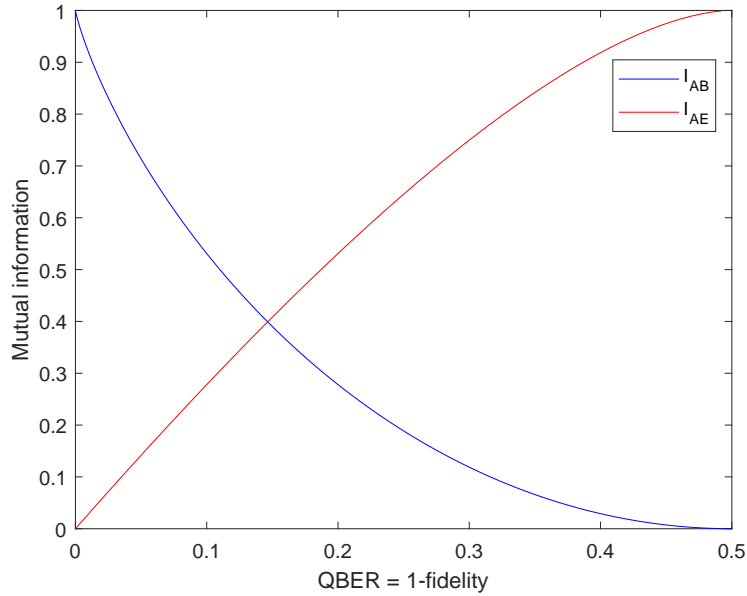


Figure 3.5 – Mutual information between Alice and Bob or Eve as a function of the QBER.

The intersection between the two curves gives the limit of security wanted as it corresponds to $I_{AB} = I_{AE}$ which leads to a bit rate equal to zero following the Csiszar-Körner theorem. The intersection is located at:

$$QBER = \frac{1 - \frac{1}{\sqrt{2}}}{2} \approx 15\% \quad (3.10)$$

As one of the goals of this thesis is to describe several discrete protocols with the same formalism and to obtain the mutual information as a function of the QBER, this graph and this limiting value are important as they will allow us to verify our results by comparing them to it. It is worth mentioning that this limit is only valid when considering individual attacks and not the other attacks. If the other attacks are considered, this value goes down to 12 %.

3.4.2 Other protocols

There exist other protocols with discrete variables that have a relatively high importance such as SARG04 [29], the protocol E91 [15], the protocol B92 [1] involving only two states... However, they are all derived from BB84.

In this work, they are less important than BB84 but E91 and SARG04 are briefly introduced:

- **E91:** It was invented by Ekert in 1991. This protocol is very similar to BB84. The main difference is the stage of the preparation by Alice. In E91, Alice does not prepare a state but has a generator of entangled states. This generator gives two quantum bits that are entangled. One of these bits is sent to Bob while Alice keeps the other. She then performs a measurement like Bob. The rest of the protocol is the same as in BB84. However, to calculate the security bounds, it might be better to use this formalism of entangled states.
- **SARG04:** This protocol was invented to overcome one of the flaws of BB84: the photon number splitting attack. Due to the fact that a source can never be perfect,

sometimes two photons in the same state can be emitted instead of one. Eve has just to keep one of the two photons. She stores it and waits for the bases reconciliation. She can then do the good measurement and obtain the totality of the information. To avoid this, SARG04 does not store data into the states but into the bases: the computational basis corresponds to 0 while the dual one corresponds to 1. A sign is assigned to each state: $|0\rangle$ and $|-\rangle$ are negative while the two others are positive. The preparation by Alice and the measure by Bob are the same. The first stage that differs is the bases reconciliation: instead of revealing the basis used, Alice reveals the sign that she obtained. If Bob has the same sign, he could have used the same basis or the wrong one but if he has another sign he is sure that he used a different basis than Alice. Only this case will be kept and Bob will take the bit that does not correspond to the basis he used. If he has the same sign, all the information are removed: they are useless.

This method prevents Eve from obtaining all the information that she had for BB84 when two bits were emitted.

3.5 Continuous variables protocols

Two difficulties of the discrete protocols are the single-photon source and the detectors that must allow to detect single photons. In protocols with continuous variables, these difficulties are avoided by using simple photodiodes as detectors. Their advantage is that they are faster and more efficient. The method used to measure will then be an homodyne detection scheme (for the most basic protocols) and the data will be supported by a continuous carrier such as the quadratures of light: x and p . x and p appear when the light is described as an oscillatory function: $x \cos(\omega t) + p \sin(\omega t)$. The quadratures x and p then totally characterize a single mode optical field. There is no more need for a single photon source. As the commutator of x and p is known:

$$[x; p] = 2i \quad \text{if } \hbar = 2, \quad (3.11)$$

it appears that the two variables do not commute. As a consequence of this fact and of the Heisenberg inequality for two conjugated variables; it is impossible to know the values of x and p simultaneously:

$$\Delta x \Delta p \geq 1 \quad (3.12)$$

Moreover, it is impossible to clone the states $|x\rangle$ and $|p\rangle$ perfectly. Indeed, the no-cloning theorem presented in chapter 4 implies that there is no procedure which allows the spy to clone the states $|x\rangle$ and $|p\rangle$ exactly. Like for qubit, if a cloner is used to clone in the basis $|x\rangle$; it will not be possible to clone in the other basis $|p\rangle$. We are then in the same conditions as for discrete variables: Eve cannot measure the two quadratures and she cannot clone the state. This means that she will never be able to obtain information without disturbing the system which can be detected by Alice and Bob.

3.5.1 Four canonical protocols

Such protocols with continuous variables have appeared after the discrete ones with a first protocol which is the equivalent of BB84. It was developed independently by two groups of researchers during the same period ([6] and [19]). After this first step in the world of continuous variables, many other protocols with continuous variables have been developed. However, there are mainly four protocols that are very important and that

will be presented here in the order of their discovery. A good review of all these protocols is presented in [8].

Another important thing that was done for continuous cryptography is the study of direct end reverse reconciliation. This allows Alice and Bob to obtain better results when the key is based on the bits of Bob.

Cerf-Lévy-VanAssche protocol

In this protocol presented in [6] that is the continuous analogue of BB84, squeezed states are used. The objective of Alice and Bob is to share a continuous secret key composed of Gaussian-distributed variables. A squeezed state is a state that has a very low variance on one quadrature while the other variance is very high. One of the values (x or p) is thus well-defined while the second one contains no information.

Two representations of a squeezed state are presented on figure 3.6. On this representation, the state is squeezed in x which means that $\Delta x \ll \Delta p$ such that $\Delta x \Delta p = 1$. The element of key that Alice will transmit to Bob is the mean value of the squeezed quadrature; in this case: $\langle x \rangle$.

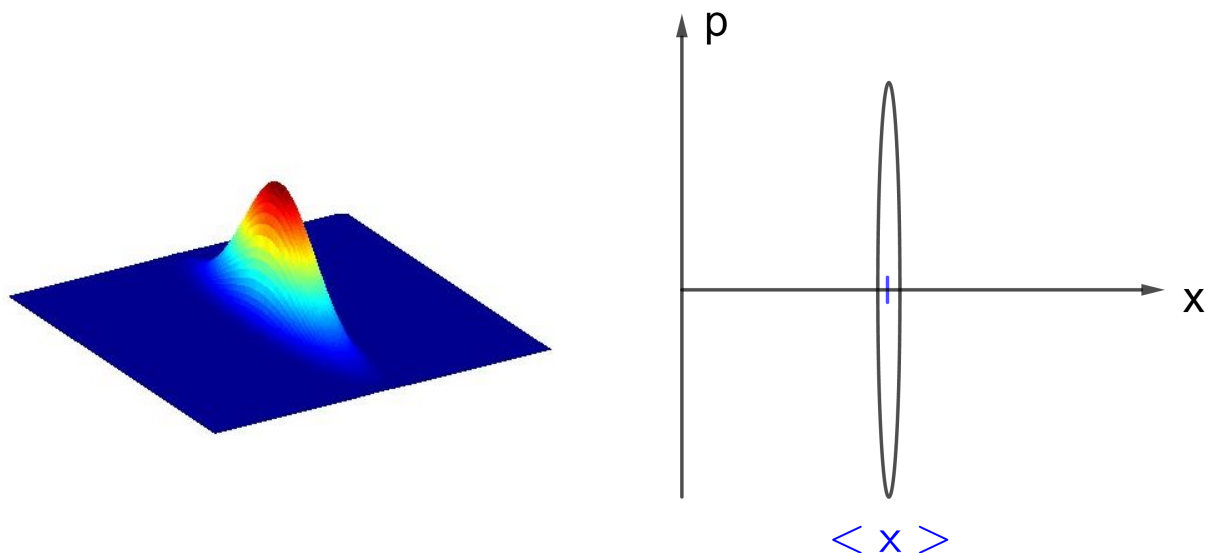


Figure 3.6 – Representation of a squeezed state. On the left: probability density distribution as a function of x and p . On the right: simple representation of a squeezed state in x .

Alice thus sends a squeezed state in x or in p . Everything is done in such a way that Eve cannot see the difference between two such states. She then does not know whether the information is stored in x or p and she cannot perform the good measurement for sure (exactly like in BB84). When Bob receives the state, he performs an homodyne measurement. This kind of measurement allows him to measure x or p and he chooses which one randomly.

When all these operations are repeated n times, Bob announces which quadratures he measured; like in BB84, if he measured the quadrature in which Alice has coded a continuous variable, they keep the information. Else, they dismiss the data to obtain a sifted key. Thereafter, the steps are the same as before: they evaluate the QBER and then they do a classical post-treatment if it is possible to obtain a secret key.

The security of this protocol has been proved for individual Gaussian attack [6] and it has also been demonstrated that protocols with continuous variables are secure against

non-Gaussian attacks [20]. However, while avoiding the use of a single photon source, this method is still not very practical due to the difficulty to generate squeezed states.

Grosshans-Grangier protocol

This was the second protocol proposed in 2002 by Grosshans-Grangier in [21]. Everything which involves Bob stays the same: he performs an homodyne measurement. The difference comes from the state sent by Alice: she sends a coherent state instead of a squeezed one. In a coherent state there is no quadrature which has a very low variance. Alice will encode both quadratures: she sends $|x + ip\rangle$. A representation of a coherent state is shown on figure 3.7.

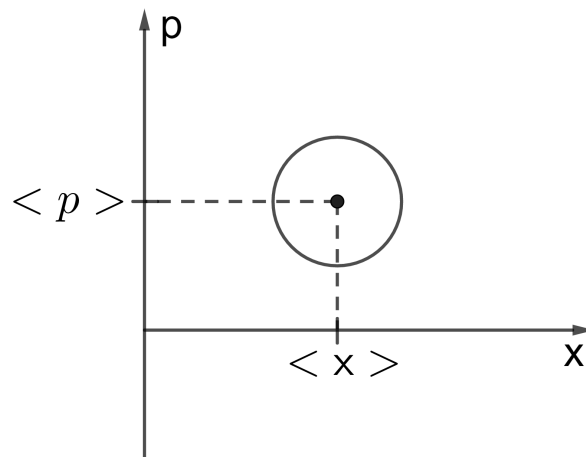


Figure 3.7 – Simple representation of a coherent state.

Due to this, there is noise on both quadratures. However, when Bob measures one quadrature, he obtains a Gaussian variable which is correlated with one of the variables of Alice. When Bob informs her of his measurement, she just throws away the variable associated with the wrong quadrature. They obtain variables which are correlated and they can detect if Eve has done anything. Indeed, if she makes a measurement and sends a new state to Bob, if she measured the wrong quadrature, Bob can see her intervention when comparing some data with Alice. Again, the security of this protocol has been demonstrated against individual Gaussian attack [22].

No basis-switching protocol

It was developed by Australian researchers in 2004 [33]. Again, Alice sends coherent states but Bob realizes heterodyne detection. This corresponds to the measure of both quadratures. He then does not have to choose randomly between two measurements which leads to the name of the protocol. This is also easier on an experimental point of view. Indeed, Bob has a single measurement device and he does not have to touch it to change the quadrature which is measured each time Alice sends him a state. To do this simultaneous measure, he uses a beam splitter and measures one quadrature at each output arm. He obtains two variables but they are noisy. Indeed, if Bob could obtain exactly the two values, the spy could do the same and obtain all the information. It has been demonstrated that an heterodyne measurement is equivalent to realizing the best cloning which is possible [9] and to measure on each clone one of the quadratures. Bob and Alice thus have continuous variables which are correlated. With the help of classical algorithms for error correction and privacy amplification, they can obtain a secret key.

Noise-tolerant protocol

The last protocol [17] was invented in 2009. This protocol is still based on squeezed states but with an heterodyne detection for Bob. Such a method may seem useless: Alice encodes a value in only one of the quadratures and Bob measures the two quadratures with more noise than if he had measured only one. When they will do bases reconciliation, Bob will eliminate the value corresponding to the wrong basis which will have as a consequence that he will have a value which is linked to the one of Alice but with more noise. When doing reverse reconciliation, this will be an advantage for Alice and Bob. Indeed, this extra noise cannot be controlled by Eve and will then be detrimental for her.

Overview of the four methods

These four methods are thus characterized by the kind of state sent by Alice and the kind of measurement made by Bob. A synthesis of these methods is presented on the table of the figure 3.8.

		Method of measurement of Bob	
		Homodyne detection	Heterodyne detection
State sent by Alice	Squeezed state	Cerf-Lévy-VanAssche protocol	Noise-tolerant protocol
	Coherent state	Grosshans-Grangier protocol	No basis-switching protocol

Figure 3.8 – Overview of the four main protocols of quantum key distribution with continuous variables.

The objective of this thesis is to obtain the same table for discrete variables and to see if better results can be obtained by using reverse reconciliation which has never been studied for discrete variables.

3.5.2 Equivalent entanglement-based protocols

A complete review of the different protocols and a proof of security in the case of reverse reconciliation is presented in [23]. To analyse the security, it is considered that Alice does not prepare a state to send it to Bob but an analogue vision is used like in discrete protocols. Instead of preparing a state, she prepares a pair of quantum states which are entangled (named EPR state following the article written by Einstein, Podolsky and Rosen [14]). On one of the states, she makes an homodyne or heterodyne measure and she sends the other one to Bob. This is totally equivalent to the case where she prepares a state due to the entanglement. Such preparation is represented on figure 3.9.

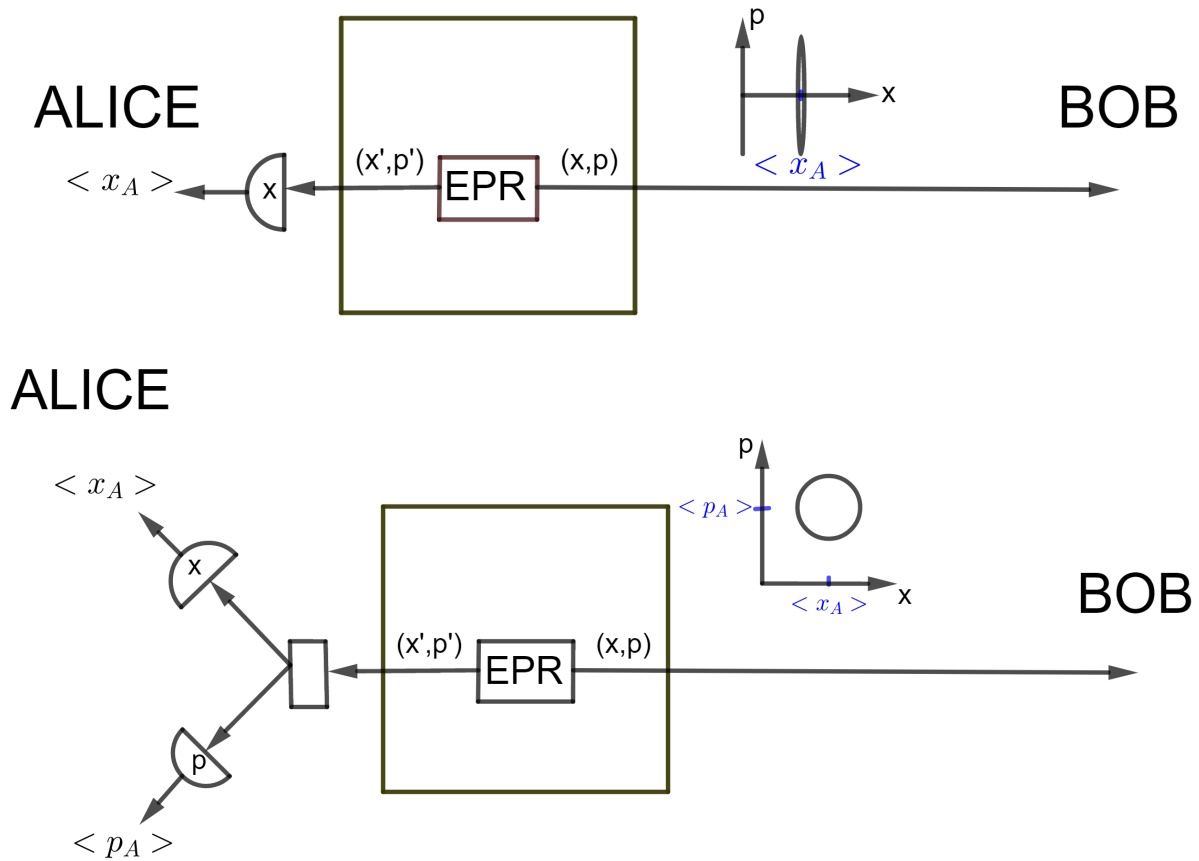


Figure 3.9 – Preparation of the state by Alice using entangled states. Above: Alice measures only the quadrature x at the output of the black box creating an EPR state. Bob then has a squeezed state with a value of the encoded quadrature centered on the value measured by Alice. Below: same representation in the case where Alice performs an heterodyne measurement. Bob obtains a coherent state.

With these preparations, it is easier to calculate if a protocol is secure. The analysis is based on the conditional variance on the value of the quadrature obtained by Bob if the measurement of Alice is known. It is then supposed that Eve uses the best possible cloning machine which is the cloner that minimises the conditional variance on the quadratures of Bob knowing the values measured by Eve. All these variances are then transformed in Shannon's entropies which are used into the Csiszar-Körner theorem to see in which limits it is possible to obtain a key and which protocol is the best.

3.6 Conclusion

In this chapter, we saw first the foundations of quantum cryptography before explaining more in detail some protocols. The more important one in the scope of this thesis is BB84 and the limit of security which is obtained by equalizing mutual information. This value will be found again in the formalism used to analyse other protocols. The idea to developed such protocols came from the various continuous protocols which explains why they were also studied in this chapter.

Chapter 4

Cloning of a quantum state

The history of quantum cloning goes back to 1982 with the article of Herbert [24]. In this article, he claims that it is possible to make communication faster than light by using an idealized laser gain tube presented in his article. This tube would produce distinguishable states of light from an incoming single photon in any polarization state due to stimulated emission. The noise should not prevent from distinguishing the polarisation state and super-fast communication would be possible. Everyone in the scientific community was perplex about this idea because it would violate the principle of causality. The reviewers decided to publish it even if they thought that it was wrong so that the scientific community could search the weakness. This is what happened: lots of people have begun to search for the weakness of Herbert's article.

At the end of the same year, it was demonstrated that it is impossible to make a perfect copy of any quantum state which was supposed possible in the paper of Herbert. This demonstration was done almost simultaneously by Wootters and Zurek [35] and by Dieks [12]. Some years after this discovery, some articles introduced the concept of imperfect cloning machine: the clones are not exactly the same as the input state. These imperfect quantum cloning machines are very important in the framework of quantum cryptography as they allow the scientists to model the attack that Eve performs. The spies can use these machines in order to obtain as much information on the input state as they can. This is thus important to present the subject in this thesis as it will be used in the proofs of security that will be established.

In the first section of this chapter, the fact that it is impossible to clone any quantum state is enunciated and demonstrated. Then, some imperfect quantum cloning machine are presented as they are used to represent the best possible methods for a spy to intercept a message. A review about quantum cloning can be found in [7].

4.1 No-cloning theorem

Cloning a quantum state (realizing one or more copies of the quantum state) is impossible due to the linearity of quantum mechanics. We will demonstrate it by using the same reasoning presented by Wootters and Zurek in their article.

If it was possible to clone a quantum state, starting from a state $|n\rangle$, the cloning procedure should correspond to:

$$|n\rangle |0\rangle |A_{in}\rangle \rightarrow |n\rangle |n\rangle |A_{out}\rangle \quad (4.1)$$

where $|A_{in}\rangle$ and $|A_{out}\rangle$ are the initial and final states of the cloning machine while $|0\rangle$ is a free ancilla that will contain the output. The ancilla is there because there must be

same number of inputs and outputs as in quantum mechanics everything is described by unitary transformations. Such procedure is represented on figure 4.1.

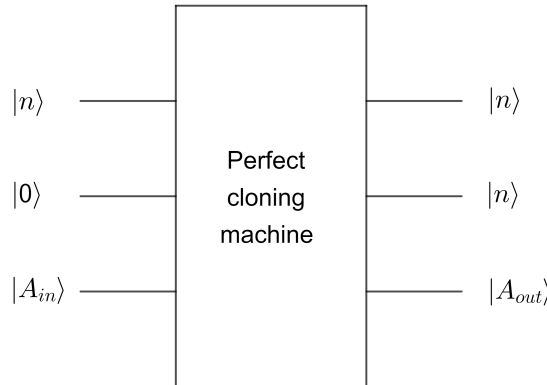


Figure 4.1 – Perfect cloning machine.

We will demonstrate that this is impossible if the state $|n\rangle$ does not belong to a well-defined basis. First, we start by considering that it is possible to clone a state which is in the computational basis $\{|0\rangle, |1\rangle\}$. This means that the action of the unitary operator corresponding to the cloning machine on these states is:

$$\begin{cases} U_{cl} |0\rangle |0\rangle |A_{in}\rangle = |0\rangle |0\rangle |A_{out}\rangle \\ U_{cl} |1\rangle |0\rangle |A_{in}\rangle = |1\rangle |1\rangle |A_{out}\rangle \end{cases} \quad (4.2)$$

This transformation corresponds to a perfect cloning if we always start from a bit in the computational basis. Now, using the same operator, we want to make a clone of the state $|+\rangle$. To calculate the effect of the unitary operator, we need to use the linearity of the operator. This leads to:

$$U_{cl} |+\rangle |0\rangle |A_{in}\rangle = U_{cl} \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) |0\rangle |A_{in}\rangle = \frac{1}{\sqrt{2}} U_{cl} |0\rangle |0\rangle |A_{in}\rangle + \frac{1}{\sqrt{2}} U_{cl} |1\rangle |0\rangle |A_{in}\rangle \quad (4.3)$$

By using what we know about the cloning machine (equation 4.2), we can rewrite the output for the state $|+\rangle$:

$$\frac{1}{\sqrt{2}} |0\rangle |0\rangle |A_{out}\rangle + \frac{1}{\sqrt{2}} |1\rangle |1\rangle |A_{out}\rangle \quad (4.4)$$

The fact is that this is not equal to the state that we want:

$$\begin{aligned} |+\rangle |+\rangle |A_{out}\rangle &= \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) |A_{out}\rangle \\ &= \left(\frac{|0\rangle |0\rangle + |1\rangle |0\rangle + |0\rangle |1\rangle + |1\rangle |1\rangle}{2} \right) |A_{out}\rangle \end{aligned} \quad (4.5)$$

It is thus impossible to perfectly clone every basis. If we want to clone the states in a known basis perfectly, it is thus possible but with this cloning machine it will be impossible to clone the states of another basis. This is known as the no-cloning theorem.

4.2 Imperfect cloning

Even if a perfect cloning is impossible, an imperfect cloning is possible. This was mentioned for the first time in an article by Mandel in 1983 [26]. He demonstrates that it is possible to obtain a copy of a quantum state which approaches the original one whatever the input polarization. However, this paper did not attract the attention of the scientists community and its interest was only discovered ten years later. In 1996, Hillery and Bužek published another paper [4] which introduced the concept of a quantum cloning machine (which was implicitly contained into the paper of Mandel).

The fact that the obtained states are not totally the same as the initial one will be characterized by a fidelity. It corresponds to the probability that the cloned state will be identified as the real state. The mathematical definition of fidelity is given by equation 4.6 for two density matrices ρ and σ . The value obtained then describes how close these two matrices are; it belongs to the interval from 0 to 1. The fidelity is equal to 0 if the two density matrices are totally different while it is equal to 1 if they are the same.

$$F(\rho, \sigma) = \text{Tr} \left(\sqrt{\sqrt{\rho} \sigma \sqrt{\rho}} \right)^2 \quad (4.6)$$

The square root appearing in this expression is the unique positive square root of the matrix if this matrix is positive semi-definite. This corresponds to the square root given by the spectral theorem. Even if it does not seem obvious from this definition, it is possible to demonstrate that the fidelity is symmetric:

$$F(\rho, \sigma) = F(\sigma, \rho) \quad (4.7)$$

This expression can be rewritten if one or the two density matrices correspond to a pure state. Indeed, if:

$$\rho = |\Psi_\rho\rangle \langle \Psi_\rho| = (|\Psi_\rho\rangle \langle \Psi_\rho|)^2, \quad (4.8)$$

the fidelity is equal to:

$$\begin{aligned} F(\rho, \sigma) &= \text{Tr} \left(\sqrt{|\Psi_\rho\rangle \langle \Psi_\rho| \sigma |\Psi_\rho\rangle \langle \Psi_\rho|} \right)^2 = \langle \Psi_\rho | \sigma | \Psi_\rho \rangle \text{Tr}(|\Psi_\rho\rangle \langle \Psi_\rho|)^2 \\ &= \langle \Psi_\rho | \sigma | \Psi_\rho \rangle \end{aligned} \quad (4.9)$$

where the last equality is due to the property of the trace of a projector ($\text{Tr}(|\Psi\rangle \langle \Psi|) = 1$).

If σ is also a pure state: $\sigma = |\Psi_\sigma\rangle \langle \Psi_\sigma|$, the fidelity can finally be rewritten as:

$$F(\rho, \sigma) = \langle \Psi_\rho | \Psi_\sigma \rangle \langle \Psi_\sigma | \Psi_\rho \rangle = |\langle \Psi_\sigma | \Psi_\rho \rangle|^2 \quad (4.10)$$

Now that we saw how to characterize the quality of the clone, we will see two cloning machines which are very important. The first one is the universal cloning machine which was presented by Hillery and Bužek in their article of 1996 while the second one is a special case of the Pauli cloning machine invented by Cerf in [5], that was presented in an article of Bruss et al in 2004 [3] and is named phase covariant quantum cloning. The universal cloning machine is very important as it is one of the first imperfect cloning machines that was invented; this is why it is briefly presented. In this work, the second cloning machine is of greater interest as it is the one that will be used to study the attack of Eve.

4.2.1 Universal cloning machine

The universal cloning machine is a machine that clones each quantum state with the same fidelity: whatever the initial state on the Bloch sphere, the output will always have the

same fidelity. This means that the two clones that come out of the cloning machine must have the same fidelity and this fidelity cannot depend on the input state: if A and B are the clones:

$$F_A(\psi_{in}) = F_B(\psi_{in}) \quad \forall \psi_{in} \quad (4.11)$$

It was demonstrated in 1998 that indeed, the cloning machine proposed by Hillery and Bužek is the optimal universal cloning machine that can be obtained. It allows the two clones to have a fidelity of $\frac{5}{6}$.

The action of the cloning machine can be expressed under the form:

$$\begin{cases} |0\rangle |C\rangle \rightarrow \sqrt{\frac{2}{3}} |00\rangle_{AB} |0\rangle_C + \sqrt{\frac{1}{3}} |\Psi_+\rangle_{AB} |1\rangle_C \\ |1\rangle |C\rangle \rightarrow \sqrt{\frac{2}{3}} |11\rangle_{AB} |1\rangle_C + \sqrt{\frac{1}{3}} |\Psi_+\rangle_{AB} |0\rangle_C \end{cases} \quad (4.12)$$

where $|C\rangle$ is an ancilla useful for the calculations as well as an ancilla that will contain the second output; $|\Psi_+\rangle$ is one of the Bell states presented in the equation 1.9. The output states A and B are the two clones while the output C is an ancilla which is useless.

Since early 2000, several experimental setups have been tested to implement this cloning machine. The more basic ones use stimulated parametric down-conversion. This is a way to obtain two entangled photons from one pump photon. This is possible due to the use of nonlinear optics. In this case the pump photon is the photon that we want to copy and the two entangled photons are the two copies. During the following years, the setup has been developed and complicated in order to obtain better results. However, the main principle is still the stimulated parametric down-conversion. On average, all these experiments did allow the scientists to measure a mean fidelity of 0,81 which is close to the theoretical value:

$$F = \frac{5}{6} \approx 0,833 \quad (4.13)$$

This means that the error is approximately equal to 2,8 % of the theoretical value which means that this experimental setup is quite good to implement a universal cloning machine. Another experimental setup is presented in [16] where only passive linear optics is used.

4.2.2 Phase-covariant cloning

This second imperfect cloning machine was proposed in 2004 by Bruss et al. In this imperfect cloning machine, all the states will not be cloned with the same fidelity: a part of the states will be cloned with an higher fidelity than the one obtained from a universal cloning machine but this is done at the cost of a lower fidelity for the states which are not in this part of the Bloch sphere.

The states that will be cloned with a higher fidelity are the ones situated on the equator of the Bloch sphere as represented on the figure 4.2.

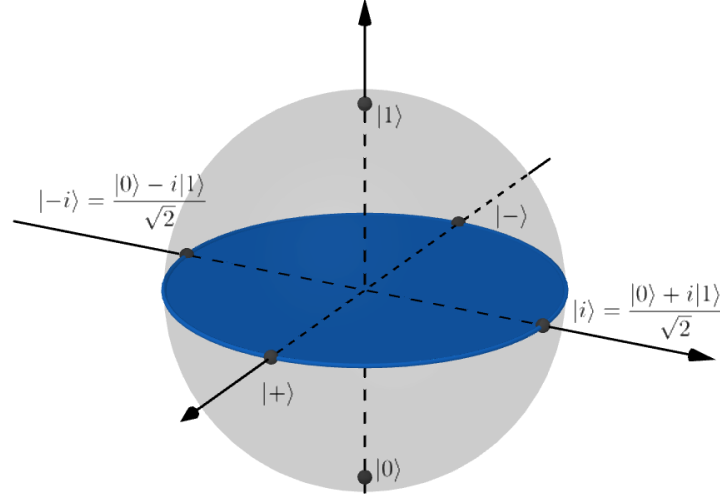


Figure 4.2 – Representation of the Bloch sphere. In blue: states that will be cloned with a higher fidelity by using a phase covariant cloning.

These qubits on the equator can be expressed by a general expression:

$$|\Psi\rangle = \frac{|0\rangle + e^{i\phi} |1\rangle}{\sqrt{2}} \quad (4.14)$$

where ϕ represents an arbitrary angle.

By using the cloning machine defined in their article, Bruss et al. demonstrated that the fidelity for any input state on the equator is equal to 0.854 which is higher than the fidelity found with a universal machine. However, it indeed appears that the possibility to obtain such higher fidelity goes with a lower fidelity for the other states of the Bloch sphere as the fidelity of the pole is only equal to $\frac{3}{4}$.

Unlike with the universal cloning machine, when defining the action of the cloning on an input state, the third input bit (an ancilla that we throw away after the cloning process) is useless. There are then just two qubits involved: the state that we want to copy and a blank ancilla to contain the second clone. This operation can be written:

$$\begin{cases} |0\rangle |0\rangle \rightarrow |0\rangle_A |0\rangle_B \\ |1\rangle |0\rangle \rightarrow \frac{1}{\sqrt{2}}(|0\rangle_A |1\rangle_B + |1\rangle_A |0\rangle_B) \end{cases} \quad (4.15)$$

When the operation is expressed under this form, this means that it is symmetric: the two clones will have the same symmetry. An important remark can be made: the phase-covariant cloning machine for states on the equator is defined with the basis of states which are orthogonal with respect to the equator.

However in quantum cryptography, the more important cloner is an asymmetric one where one of the copies can be better than the other. The spy can then have more or less information about the state sent by Alice than Bob. To define an asymmetric phase-covariant cloning machine, it is enough to break the symmetry between the superposition of $|01\rangle$ and $|10\rangle$ at the output of the machine:

$$\begin{cases} |0\rangle |0\rangle \rightarrow |0\rangle_A |0\rangle_B \\ |1\rangle |0\rangle \rightarrow \cos(\theta) |0\rangle_A |1\rangle_B + \sin(\theta) |1\rangle_A |0\rangle_B \end{cases} \quad (4.16)$$

θ is the angle of the cloning machine which varies between 0 and $\frac{\pi}{2}$. If we use this machine, we obtain two copies of the input state with different fidelity but the fidelities will stay

the same for all the states on the equator. The fidelity of the two copies can be calculated with the equation 4.10:

$$\begin{cases} F_A = \frac{1}{2}(1 + \sin(\theta)) \\ F_B = \frac{1}{2}(1 + \cos(\theta)) \end{cases} \quad (4.17)$$

This asymmetric way of cloning the state is the one that will be used later in this work.

4.3 Conclusion

In this chapter, we saw the proof that it is impossible to clone any quantum state. Thereafter, we presented a new kind of cloning, the imperfect one which was invented to "replace" a perfect cloning machine. To characterize such operations, there is a well-known variable which is the fidelity. Finally, we saw two important examples of such machines that will be useful for this work.

Part II

Results

Chapter 5

Objective of the research

As said before, the first protocol proposed to do quantum cryptography involves discrete variables: BB84. After that, a huge step has been done with the discovery of continuous variables protocols. Indeed, these last ones are more easy to apply experimentally than those using single photons. The first protocol with continuous variables that was defined is the analogue of BB84, the Cerf-Lévy-VanAssche protocol, but many others were found after that. However, nobody has never done the reverse thinking: starting from these protocols with continuous variables and searching for their discrete analogues. This is the objective here: finding the total analogue of the table 3.8.

Moreover, protocols with discrete variables have only been studied for direct reconciliation, but in the continuous variables case, it has been demonstrated that using reverse reconciliation could allow Alice and Bob to have a protocol which is more robust. It can raise a question: in discrete variables, could reverse reconciliation allow for better results than the ones obtained with BB84?

This leads to the two main points that will be presented in this part:

- **Research of the analogous version of the four main protocols with continuous variables:** To find these new protocols and to study them we first need to define each step that is involved in the procedure. Thereafter, the security will be studied to see if it is possible to obtain better results than those of BB84.
- **Study of the reverse reconciliation in the case of discrete variables:** All the protocols defined before must be modified. The objective of Eve will change, she does not want to obtain information on Alice's bits anymore but she wants to find the ones of Bob.

These two points will be explored in the two next chapters: the first one is about the definition of the protocols and their study in the case of direct reconciliation. In the second one, reverse reconciliation will be studied. Thereafter, a conclusion is done in the last chapter.

Chapter 6

Security of the four canonical protocols supplemented with direct reconciliation

6.1 Introduction

In this chapter, we will study the four discrete protocols corresponding to the ones presented on figure 3.8. Only the direct reconciliation will be considered as the reverse one is the subject of the following chapter.

This chapter is divided in three parts. In the first one, we begin by defining the four protocols that will be studied. These four protocols will be named in this work according to the names of the analogous protocols with continuous variables. The names used are therefore: BB84, discrete Grosshans-Grangier protocol, discrete no basis-switching protocol and discrete noise-tolerant protocol.

In the second section, the different stages such as the preparation of the states, the action of Eve,... will be described in terms of density matrix, POVM,... The objective with this formalism will be to find the expression of the joint probability of a certain output of the measurement for Bob and Eve conditionally on the input of Alice.

This result will be used in the last section to determine the security of the protocol and to compare the three new protocols to BB84.

6.2 Description of the protocols

As we saw in chapter 3, the four protocols differentiate themselves in two ways: the measurement done by Bob and the kind of states sent by Alice. This is the reason why we will start by defining precisely the states that Alice can send and the measurement that Bob can do. The last point of this section will be dedicated to the action of Eve and how it can be represented with a cloning machine.

6.2.1 States sent by Alice

For continuous variables, there were two kinds of states that Alice could send: squeezed or coherent states. We need to find the analogous version of these two states to be able to define correctly the discrete protocols.

We already know what is the equivalent of the squeezed states sent by Alice. Indeed, we said that the first protocol with continuous variables that was invented uses squeezed states and is the continuous analogue of BB84. We then know that the equivalent of the squeezed states for Alice are the states in BB84. For reminder, the states involved in BB84 are the ones of the computational basis and of the dual one: $|0\rangle$, $|1\rangle$, $|+\rangle$ and

$|-\rangle$. With a probability equal to one half, she encodes a random bit in the computational basis and with the same probability, she encodes it in the dual basis. To represent these four states, we will use a part of the Bloch sphere: the vertical plane that contains these four states. They are represented on figure 6.1. The fact that Alice encodes with an equal probability in one or the other basis in discrete protocols is exactly the analogue to the fact that in squeezed states, she encodes only one of the quadratures (with the same probability for each quadrature).

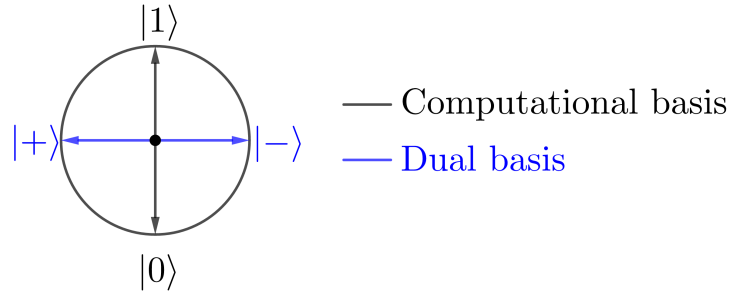


Figure 6.1 – Representation of the analogue of the squeezed states that Alice can send.

These states are thus the ones that will be sent for BB84 and for the discrete noise-tolerant protocol.

For the two other protocols, we need to find the equivalent of coherent states. In such states, Alice has encoded the two quadratures. In discrete, it would correspond to the fact that she sends a state encoded in the computational and the dual bases simultaneously. On the considered circle of the Bloch sphere, this corresponds to the "diagonal" states. Alice can then send one state among the four possible. To each of them, two bits can be associated. The figure 6.2 presents such states and the associated bits. In this protocol, there is no question about which basis to use. Indeed, Alice has just to send one of the four states with an equal probability ($\frac{1}{4}$ for each state).

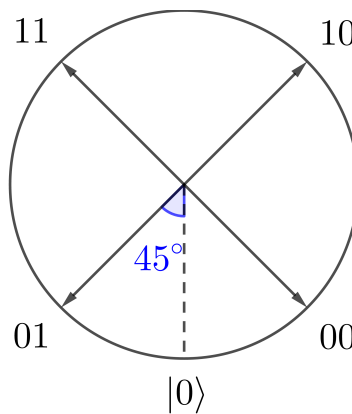


Figure 6.2 – Representation of the analogue of the coherent states that Alice can send.

An important remark for the calculation that will be made in the following section is that all these states lie on the same plane of the Bloch sphere: the vertical one that contains the states of the computational basis and of the dual basis.

6.2.2 Measurement done by Bob

Again, in the continuous protocols, two kinds of measurements were done: the homodyne detection and the heterodyne one.

The homodyne detection is just the measurement of one of the two quadratures of the light. In our discrete framework, it is thus equivalent to the measurement in one of the two bases. This is exactly what was already used by Bob in BB84: he measures randomly in one of the two bases and then he compares this basis to the one of Alice. Such a measurement is a projective measurement as presented in chapter 1 on one of the two bases represented on the figure 6.1.

To obtain all the different elements that will allow us to study the four protocols, we need to define a second kind of measurement which is analogous to heterodyne detection. In these measurements performed by Bob, the two quadratures are measured with more noise than in an homodyne one. In our case, this means that we need to measure in the two bases simultaneously. To do this a POVM will be used. This POVM will be constituted of four projectors on the four states which are presented on the figure 6.2.

When we put together the kind of states sent by Alice and the measurements done by Bob, we obtain a full view of the four protocols. This is presented on the figure 6.3. This table is equivalent to the one seen in the continuous case.

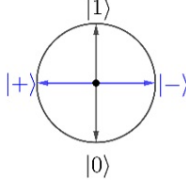
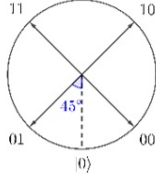
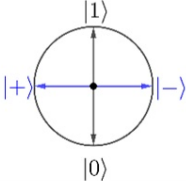
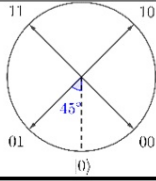
		Method of measurement of Bob	
		Projective measurement on one of the two basis (randomly chosen)	POVM constituted of the four states
			
State sent by Alice		BB84	Discrete noise-tolerant protocol
		Discrete Grosshans-Grangier protocol	Discrete no basis-switching protocol

Figure 6.3 – Overview of the four discrete protocols that are studied.

6.2.3 Action of Eve

At this stage, we know for each protocol what Alice and Bob will do. We still need to know what Eve will do. Like in all the protocols, she will interact with the quantum channel. Her objective is to obtain as much information on Alice's state as she can without modifying the state too much. Otherwise, she would immediately be detected by Bob. To do so, she will use a quantum cloning machine. As explained in the chapter 4, it is impossible for her to make a perfect copy of the state. However, she can make imperfect copies. As mentioned above, in all the protocols that are considered here, only states of

one plane of the Bloch sphere are involved. To do the best cloning possible on a plane of the Bloch sphere, Eve can use the phase covariant cloner described in the chapter about cloning. As a reminder, we can rewrite the effect of this cloner:

$$\begin{cases} |0\rangle_A |0\rangle \rightarrow |0\rangle_B |0\rangle_E \\ |1\rangle_A |0\rangle \rightarrow \cos(\alpha) |0\rangle_B |1\rangle_E + \sin(\alpha) |1\rangle_B |0\rangle_E \end{cases} \quad (6.1)$$

where α is the angle of the cloning machine. B and E represent Bob and Eve that will each receive one of the clones while A is the bit of Alice. The angle of the cloning machine varies between 0° where the state of Alice is sent to Eve and 90° where the state of Alice is sent to Bob. However, if we keep this definition of the cloner, it represents a cloning machine that clones with a higher fidelity all the states that are in the equatorial plane of the Bloch sphere than the ones out of it, but this is not what we want. We want to clone the states on the vertical plane. These two different planes are presented on figure 6.4.

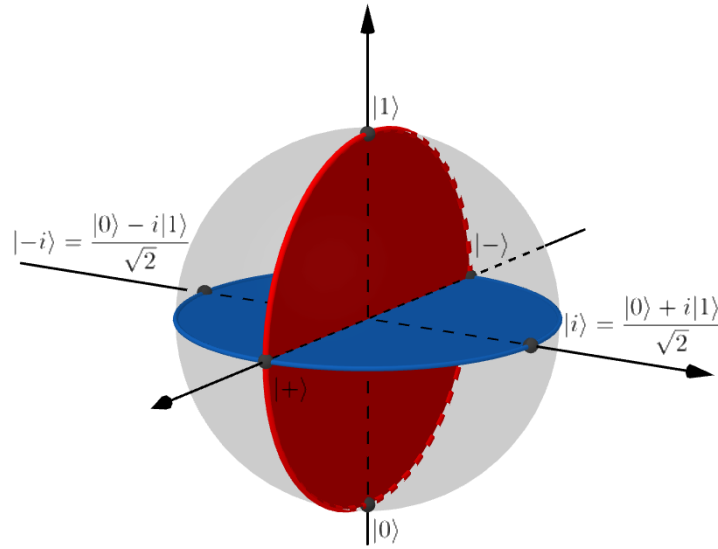


Figure 6.4 – Representation of the Bloch sphere. In blue: states that will be cloned with a higher fidelity by using the phase covariant cloning just enunciated. In red: states that we want to clone with the highest fidelity.

It is necessary to change the definition of the cloning machine to clone the right plane. It is clear that to pass from the blue situation on the scheme to the red one, we need to do a rotation. To clone the blue circle, the cloner was defined as a function of the computational basis which is composed of the states that are orthogonal to the plane on the sphere. It is thus obvious that if we want to optimally clone the red circle, we need to use the same machine by replacing the computational basis by the basis $\{|i\rangle, |-i\rangle\}$. This leads to the expression of the cloning machine used by Eve:

$$\begin{cases} |i\rangle_A |i\rangle \rightarrow |i\rangle_B |i\rangle_E \\ |-i\rangle_A |i\rangle \rightarrow \cos(\alpha) |i\rangle_B |-i\rangle_E + \sin(\alpha) |-i\rangle_B |i\rangle_E \end{cases} \quad (6.2)$$

Eve can use this cloner to obtain a copy while she sends the other to Bob. On her copy she can then make the same measurement as Bob to obtain information on the states of Alice.

6.3 Mathematical description of the different stages

In this section, we will develop mathematically the different protocols presented above. This will be done by using the different elements of quantum mechanics that were presented in chapter 1. The objective is here to do a general calculation containing parameters so that the final results can be used to describe any of the protocols by simply replacing the parameters by the appropriate values. The calculations presented in this section were partly done with Mathematica. The code can be found in annex A.

To do so, we need to remind that any state on that plane of the Bloch sphere can be rewritten as:

$$|\Psi\rangle = \cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) |1\rangle \quad (6.3)$$

This equation is immediately derived from the expression 1.4. Indeed, they are identical except for the factor of the phase. This factor can be removed here as we limit ourselves to the plane: all the imaginary combinations lead to points that are out of it.

As θ is the angle on the Bloch sphere, we can easily find what angle will correspond to the four states used in BB84:

- $|0\rangle$: $\theta = 0^\circ$
- $|1\rangle$: $\theta = 180^\circ$
- $|+\rangle$: $\theta = 90^\circ$
- $|-\rangle$: $\theta = 270^\circ$

This can be proved by injecting the various angles in the expression. For example, for the state $|+\rangle$, it gives:

$$|\Psi\rangle = \cos\left(\frac{90^\circ}{2}\right) |0\rangle + \sin\left(\frac{90^\circ}{2}\right) |1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \equiv |+\rangle \quad (6.4)$$

For the four other states used in the other protocols, the corresponding angles will then be 45° , 135° , 225° and 315° .

Alice will send a state of this form and when we will analyse the different protocols, we will replace the angle θ by different values according to the considered protocol.

The next stage in this scheme is the cloning by Eve. It is necessary to apply the equation presented before (6.2). However, before doing that, we have to change the basis as the state of Alice is in the computational basis and the cloning machine is presented for another basis.

In order to change the basis, let us first remind the definition of the states $|i\rangle$ and $|-i\rangle$ as a function of the computational basis. These equations can then be inverted to find the inverse relation: the expression of the states of the computational basis as a function of the basis $\{|i\rangle, |-i\rangle\}$. Such inversion is done in the equation 6.5.

$$\begin{cases} |i\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \\ |-i\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \end{cases} \Leftrightarrow \begin{cases} |0\rangle = \frac{1}{\sqrt{2}}(|i\rangle + |-i\rangle) \\ |1\rangle = \frac{1}{\sqrt{2}i}(|i\rangle - |-i\rangle) \end{cases} \quad (6.5)$$

With these expressions, it is possible to express the state in the right basis:

$$\begin{aligned}
 |\Psi\rangle &= \cos\left(\frac{\theta}{2}\right) |0\rangle + \sin\left(\frac{\theta}{2}\right) |1\rangle \\
 &= \cos\left(\frac{\theta}{2}\right) \frac{1}{\sqrt{2}}(|i\rangle + |-i\rangle) + \sin\left(\frac{\theta}{2}\right) \frac{1}{\sqrt{2}i}(|i\rangle - |-i\rangle) \\
 &= \frac{1}{\sqrt{2}}\left\{\left(\cos\left(\frac{\theta}{2}\right) - i\sin\left(\frac{\theta}{2}\right)\right)|i\rangle + \left(\cos\left(\frac{\theta}{2}\right) + i\sin\left(\frac{\theta}{2}\right)\right)|-i\rangle\right\} \\
 &= \frac{1}{\sqrt{2}}\{e^{-i\frac{\theta}{2}}|i\rangle + e^{i\frac{\theta}{2}}|-i\rangle\}
 \end{aligned} \tag{6.6}$$

With the state expressed under this form and an added ancilla, it is possible to apply the cloning machine:

$$|\Psi\rangle |i\rangle \xrightarrow{\text{cloning}} |\Psi_{\text{clone}}\rangle = \frac{1}{\sqrt{2}}\{e^{-i\frac{\theta}{2}}|i\rangle|i\rangle + e^{i\frac{\theta}{2}}(\cos(\alpha)|i\rangle|-i\rangle + \sin(\alpha)|-i\rangle|i\rangle)\} \tag{6.7}$$

In this new state, there is one bit that will go to Bob (the first one) and the other one will go to Eve. If we reformulate the state in the formalism of density matrix, by doing the partial trace, we will be able to obtain the density matrix of Bob and the one of Eve. Moreover, this is easier to manipulate as the state is now linked to a matrix and the application of an operator on it will lead to matrix multiplication.

To find the matrix density, we will apply the formula 1.8. Here, there is only one term in the sum which gives:

$$\rho_{BE} = |\Psi_{\text{clone}}\rangle \langle \Psi_{\text{clone}}| = \begin{pmatrix} \frac{1}{2} & \frac{1}{2}e^{-i\theta}\cos(\alpha) & \frac{1}{2}e^{-i\theta}\sin(\alpha) & 0 \\ \frac{1}{2}e^{i\theta}\cos(\alpha) & \frac{\cos^2(\alpha)}{2} & \frac{1}{2}\cos(\alpha)\sin(\alpha) & 0 \\ \frac{1}{2}e^{i\theta}\sin(\alpha) & \frac{1}{2}\cos(\alpha)\sin(\alpha) & \frac{\sin^2(\alpha)}{2} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{6.8}$$

$\langle \Psi_{\text{clone}}|$ is obtained by taking the complex conjugate of the $|\Psi_{\text{clone}}\rangle$ if they are formulated under a vectorial form.

At this stage, we have the expression of the density matrix for Bob and Eve. In the protocol, there is only one remaining step: the measurement. We want to find the probability that Bob and Eve will have some output knowing the state sent by Alice. Such probability is given by the equation 1.19 where ρ is the density matrix that we have just found. We just need to express the operators M_m . In this case, we can express it more in details as Bob and Eve both perform a measurement on their own qubit. This means that the measurement operator can be expressed as a tensor product between the measurement operators of Bob and Eve:

$$p_{\theta_b, \theta_e} = \text{Tr}(\rho_{BE} M_{\theta_b} \otimes M_{\theta_e}) \tag{6.9}$$

We put the angle θ_b and θ_e in anticipation. Indeed, we will see that the different elements of a POVM will differentiate themselves by an angle.

Again, the idea is to formulate it the same way if Bob and Eve both perform a projective measurement (analogue to homodyne detection) or a POVM (analogue to the heterodyne detection). These two kinds of measurement were presented in section 1.5. The projective measurements can be interpreted in terms of POVM. Indeed, Bob will

measure either in the computational basis or in the dual basis with a probability of $\frac{1}{2}$. The projective measurement in one of the two bases can be expressed as a POVM containing two elements which are the projectors on the two states forming the basis. For example, for the computational basis:

$$\text{POVM} = \{|0\rangle\langle 0|, |1\rangle\langle 1|\} \quad (6.10)$$

Such a POVM fulfils the conditions: each element is positive and their sum is equal to the identity according to the closure relationship. The projective measurements done in BB84 can then be seen as two POVMs each applied with a probability of one half.

The only need to describe the measurements is then to describe the elements of a POVM. In our protocol, the elements of a POVM are always projectors on different states of the vertical plane of the Bloch sphere. In order to keep the generality for the four protocols, we will describe them using a parametric angle like we did for Alice. Again, we express the state using the expression in the right basis (6.6):

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\{e^{-i\frac{\theta}{2}}|i\rangle + e^{i\frac{\theta}{2}}|-i\rangle\} \quad (6.11)$$

The projector on this state expressed under the form of a matrix is:

$$|\Psi\rangle\langle\Psi| = \frac{1}{2} \begin{pmatrix} 1 & e^{-i\theta} \\ e^{i\theta} & 1 \end{pmatrix} \quad (6.12)$$

With this projector, we can express the POVM of Bob like an ensemble of such elements with a coefficient in front of them:

$$\text{POVM}_{Bob} = \left\{ \frac{c_b}{2} \begin{pmatrix} 1 & e^{-i\theta_b} \\ e^{i\theta_b} & 1 \end{pmatrix} \right\} \quad (6.13)$$

where θ_b is the angle corresponding to the states on which we want to project. The constant c_b is there to ensure the condition $\sum_m E_m = \mathbb{1}$ on the POVM. According to the kind of measurement done by Bob, there are two cases:

- **Measure with a POVM** : Bob's POVM contains four projectors corresponding to the elements with $\theta_b = 45, 135, 225$ and 315° . Moreover, as these four states can be seen as the states of two bases and the fact that we know that if we sum the projectors of two states of a same basis, this gives the identity (closure relationship); it is clear that c_b must be equal to $\frac{1}{2}$.
- **Projective measurement**: Bob can apply two POVMs and he uses each with a probability of one half. The first POVM corresponds to $\theta_b = 0$ and 180° while the second corresponds to $\theta_b = 90$ and 270° . Each time, the POVM contains only the two projectors on the states of the same basis. The sum of the elements is then equal to the identity and the coefficient c_b is equal to 1.

The POVM of Eve is exactly the same. We now have all the elements to express the probability that we search. We begin by calculating the tensor product by using the formula 1.13:

$$\begin{aligned} M_{\theta_b} \otimes M_{\theta_e} &= \frac{c_b}{2} \begin{pmatrix} 1 & e^{-i\theta_b} \\ e^{i\theta_b} & 1 \end{pmatrix} \otimes \frac{c_e}{2} \begin{pmatrix} 1 & e^{-i\theta_e} \\ e^{i\theta_e} & 1 \end{pmatrix} \\ &= \frac{c_b c_e}{4} \begin{pmatrix} 1 & e^{-i\theta_e} & e^{-i\theta_b} & e^{-i(\theta_b+\theta_e)} \\ e^{i\theta_e} & 1 & e^{i(\theta_e-\theta_b)} & e^{-i\theta_b} \\ e^{i\theta_b} & e^{i(\theta_b-\theta_e)} & 1 & e^{-i\theta_e} \\ e^{i(\theta_b+\theta_e)} & e^{i\theta_b} & e^{i\theta_e} & 1 \end{pmatrix} \end{aligned} \quad (6.14)$$

If we inject this into the equation 6.9, it will lead us to the probability we were searching for. This calculation implies the use of a trace which was defined in chapter 1.

$$\begin{aligned}
p_{\theta_b, \theta_e} &= \text{Tr}(\rho_{BE} \cdot M_{\theta_b} \otimes M_{\theta_e}) \\
&= \frac{1}{4} c_b c_e (\sin(\alpha) \cos(\theta - \theta_b) + \cos(\alpha) \cos(\theta - \theta_e) + \sin(\alpha) \cos(\alpha) \cos(\theta_b - \theta_e) + 1)
\end{aligned} \tag{6.15}$$

This probability is what we want: the probability that Bob and Eve will measure θ_b and θ_e if Alice did sent the state with an angle θ .

By using this distribution of probability and Shannon's information theory presented in chapter 2, we will search the security of these protocols and the achievable bit rate.

6.4 Results and security of the different protocols

6.4.1 Calculation of the mutual information

To study the security of the protocol, we want to verify for which values of the cloning angle the Csiszar-Körner theorem (equation 2.13) is satisfied. We thus need to calculate the mutual information between Alice and Bob but also between Alice and Eve. To do that, the calculation were made by using Matlab. The code can be found in annex A.

We will use the equations that were presented in chapter 2. First, we need to have the joint probability distribution between Alice, Bob and Eve. To obtain it, we can follow the equation 2.7 that links a conditional probability to a joint probability.

Here, we have the probability of Bob and Eve conditional to the state of Alice. To obtain the joint probability between the three, we need to multiply it by the probability that Alice sends this state. Again, this can have two values as a function of the state sent by Alice:

- **Analogue of a coherent state:** Alice will send one of the four state that we presented before (each with a probability of $\frac{1}{4}$).
- **Analogue of a squeezed state:** Alice chooses randomly one of the two bases. In this basis there are two states. As the problem is completely symmetric between the two bases, we will limit ourselves to the study of one of the two bases. The probability for Alice to send one of the states is then equal to $\frac{1}{2}$.

Once we have the joint probability, $p(\theta_a, \theta_b, \theta_e)$, we can calculate all the probabilities that will be used to calculate the entropies which will be involved in the calculation of the mutual information:

$$\left\{ \begin{aligned} p(\theta_a, \theta_b) &= \sum_{\theta_e} p(\theta_a, \theta_b, \theta_e) \\ p(\theta_a, \theta_e) &= \sum_{\theta_b} p(\theta_a, \theta_b, \theta_e) \\ p(\theta_a) &= \sum_{\theta_b} p(\theta_a, \theta_b) \\ p(\theta_b) &= \sum_{\theta_e} p(\theta_a, \theta_b) \\ p(\theta_e) &= \sum_{\theta_a} p(\theta_a, \theta_e) \end{aligned} \right. \tag{6.16}$$

The sum on the angles always involves the angles that correspond to the protocol that is studied.

The next stage in order to calculate the mutual information is the calculation of the entropies:

$$\begin{cases} H(Alice, Bob) = -\sum_{\theta_a, \theta_b} p(\theta_a, \theta_b) \log_2(p(\theta_a, \theta_b)) \\ H(Alice, Eve) = -\sum_{\theta_a, \theta_e} p(\theta_a, \theta_e) \log_2(p(\theta_a, \theta_e)) \\ H(Alice) = -\sum_{\theta_a} p(\theta_a) \log_2(p(\theta_a)) \\ H(Bob) = -\sum_{\theta_b} p(\theta_b) \log_2(p(\theta_b)) \\ H(Eve) = -\sum_{\theta_e} p(\theta_e) \log_2(p(\theta_e)) \end{cases} \quad (6.17)$$

With all this information, it is easy to obtain the mutual information with the formula 2.12:

$$\begin{cases} I_{AB} = H(Alice) + H(Bob) - H(Alice, Bob) \\ I_{AE} = H(Alice) + H(Eve) - H(Alice, Eve) \end{cases} \quad (6.18)$$

The mutual information is usually represented as a function of the QBER which is itself linked to the fidelity between Alice and Bob (QBER = 1-fidelity). This fidelity was given in equation 4.17:

$$F_{AB} = \frac{1 + \sin(\alpha)}{2} \quad (6.19)$$

We can demonstrate this expression in the used formalism. We will use the formula 4.9 to calculate the fidelity. In this equation, Ψ_ρ is the pure state owned by Alice while σ is the density matrix of the mixed state obtained by Bob after the cloning done by Eve.

We already have the state owned by Alice which is presented in equation 6.6. We also have the density matrix of Bob and Eve in the equation 6.8. From this density matrix, it is possible to find the density matrix of Bob by doing a partial trace as expressed in chapter 1:

$$\rho_B = \text{Tr}_E(\rho_{BE}) = \frac{1}{2} \begin{pmatrix} 1 + \cos^2(\alpha) & \sin(\alpha)e^{-i\theta} \\ \sin(\alpha)e^{i\theta} & \sin^2(\alpha) \end{pmatrix} \quad (6.20)$$

To make the calculation, we will write the state of Alice under a vectorial form:

$$|\Psi_A\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} e^{-i\frac{\theta}{2}} \\ e^{i\frac{\theta}{2}} \end{pmatrix} \quad (6.21)$$

To calculate the fidelity, we will also need the transpose-conjugate of this state:

$$\langle\Psi_A| = \frac{1}{\sqrt{2}} \begin{pmatrix} e^{i\frac{\theta}{2}} & e^{-i\frac{\theta}{2}} \end{pmatrix} \quad (6.22)$$

The fidelity is then given by:

$$\begin{aligned}
 F_{AB} &= \langle \Psi_A | \rho_B | \Psi_A \rangle \\
 &= \frac{1}{\sqrt{2}} \begin{pmatrix} e^{i\frac{\theta}{2}} & e^{-i\frac{\theta}{2}} \end{pmatrix} \cdot \frac{1}{2} \begin{pmatrix} 1 + \cos^2(\alpha) & \sin(\alpha)e^{-i\theta} \\ \sin(\alpha)e^{i\theta} & \sin^2(\alpha) \end{pmatrix} \cdot \frac{1}{\sqrt{2}} \begin{pmatrix} e^{-i\frac{\theta}{2}} \\ e^{i\frac{\theta}{2}} \end{pmatrix} \\
 &= \frac{1 + \sin(\alpha)}{2}
 \end{aligned} \tag{6.23}$$

We retrieved the theoretical fidelity that was expected.

Now, we have everything to trace the same graph as the one presented on the figure 3.5 for BB84. Moreover, here, by simply changing the parameters, we can study the four protocols.

6.4.2 Graphs and analysis

Analysis of BB84 and comparison with the theory Before studying the new protocols, a verification can be done by comparing the theoretical graph for BB84 (presented in figure 3.5) with the one obtained by using the equation developed in the previous section. The graph obtained is presented in figure 6.5.

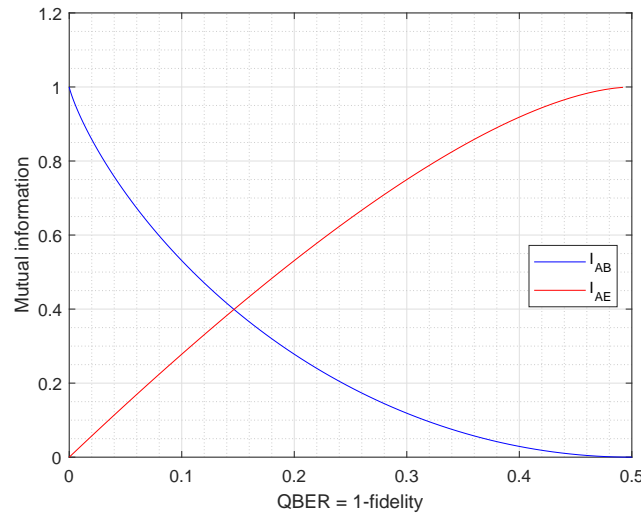


Figure 6.5 – Mutual information between Alice and Bob or Eve as a function of the QBER.

This graph shows a limit of security which is around 15 % like it was expected with the theoretical curve. This graph can be compared with the theoretical one of the figure 3.5. By doing this comparison, it appears that the two curves are exactly the same which proves that the formalism used to describe the protocol is coherent with the previous results.

Analysis of the joint probability An analysis that can be done to observe what really happens when the angle of the cloning machine is changed is the analysis of the joint probability. We have the probability in the equation 6.15. As explained before, the joint probability between Alice, Bob and Eve is obtained by multiplying the probability

of Alice to send one state with the conditional probability that was obtained before. For BB84, this probability is equal to $\frac{1}{2}$:

$$p_{\theta_a, \theta_b, \theta_e} = \frac{1}{2} p_{\theta_b, \theta_e} \quad (6.24)$$

From this, the joint probability between Alice and Bob is found by summing on the angle of Eve and the one between Alice and Eve by summing on the angle of Bob.

If we fix the angle of the state sent by Alice, we can then represent the probability for Bob or Eve to measure a certain angle. By doing this, we can see the change between the probability when we pass from an extreme angle (0°) of the cloning machine to the other extreme (90°). We already know that the fidelity between Alice and Bob is equal to $\frac{1+\sin(\alpha)}{2}$. When the angle of the cloning machine is equal to 90° , the fidelity of Bob is thus maximum while it is minimal when the angle is 0° . For Eve, it is the reverse. Here, we trace the probability for Bob and Eve to measure a certain angle. We trace it if Alice sends a state $|0\rangle$ for the two extreme angles of the cloning machine. It is presented on the figure 6.6.

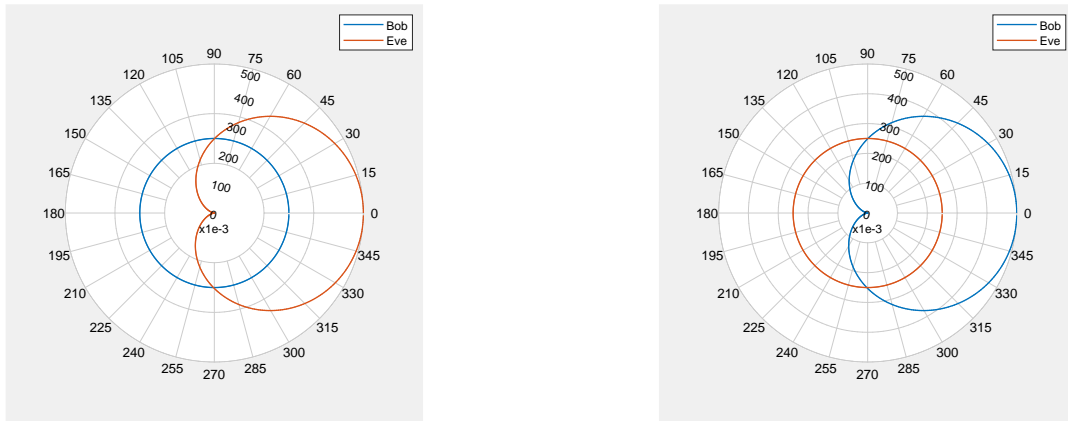


Figure 6.6 – Probability for Bob and Eve to obtain a certain angle if Alice sent the state $|0\rangle$ corresponding to an angle 0° . On the left: angle of the cloning machine equal to 0° . On the right: angle of the cloning machine equal to 90° .

On these two graphs, the result that appears is exactly what was expected: for one extreme angle of the cloning machine (on the left), Eve obtains a good clone. Indeed, she has a biggest probability to find the good angle (the one sent by Alice) than any other angle while Bob can find any angle with the same probability. Bob has then no information on the state of Alice. For the other extreme angle, it is the reverse. It also appears that the two graphs are exactly the same with the role of Bob and Eve reversed.

One can wonder how does the probability evolve between the two extreme angles of the cloning machine. This evolution is presented on the scheme 6.7. This figure represents the probability of Bob to measure the different angles in the case where Alice sent the state $|0\rangle$. The different lines correspond to different angles of the cloning machine.

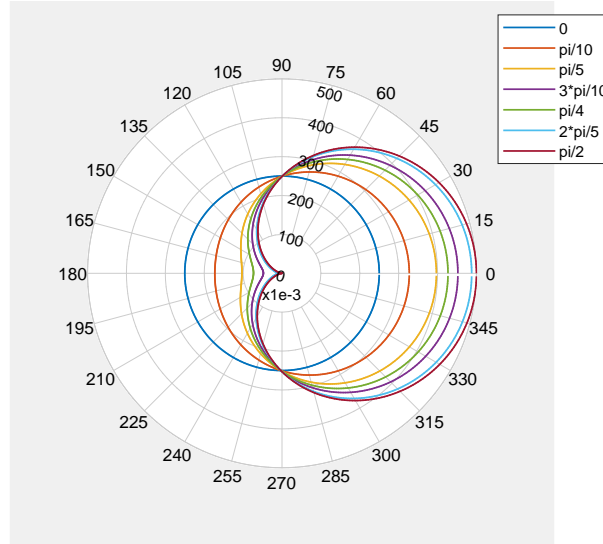


Figure 6.7 – Probability for Bob to measure the angles as a function of the angle of the cloning machine if Alice sent the state $|0\rangle$.

The two extreme lines were already visible on the graphs presented before. Between them, we can see the evolution of the probability.

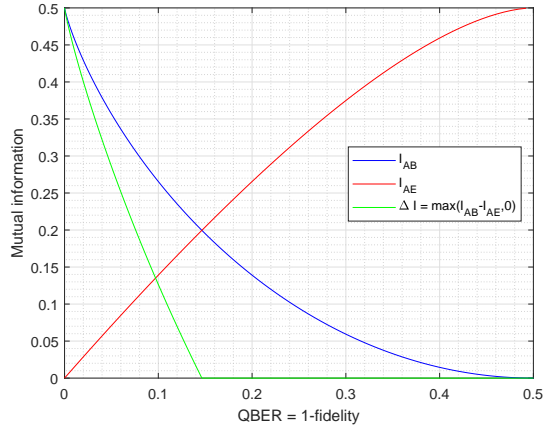
Comparison of the four protocols The objective is to compare the four protocols proposed in order to see if one of them could be better than BB84. Two features are particularly observed. The first one is the QBER at which the two mutual information are the same. This is the limit of the security of the protocol. If the QBER is bigger than this one, no private communication can be established between Alice and Bob. The expectation would be to find a protocol that has a limit which is bigger than the one of BB84.

The second element that must be observed is the bit rate which is the difference between the two mutual information. The objective is once again to obtain a rate which is bigger than the one of BB84.

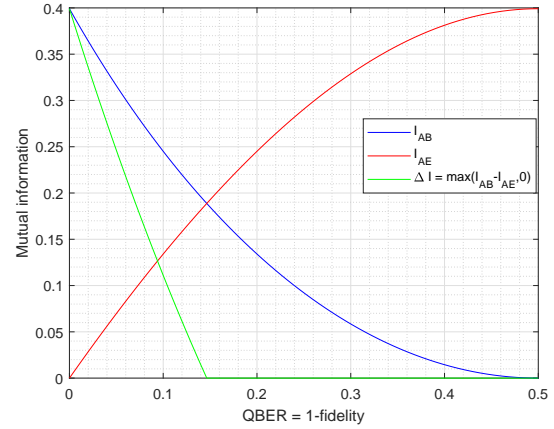
In order to be in the good conditions to compare the different graphs, we need to divide the mutual information by two in the case of BB84. Indeed, we only studied one basis over the two as it is symmetric. However, when taking into account the two bases, we must not forget that there is the stage of bases reconciliation where Alice and Bob throw their data away if they did not use the same basis.

We will then divide the mutual information by two for BB84. The four graphs obtained for the four protocols are presented on figure 6.8.

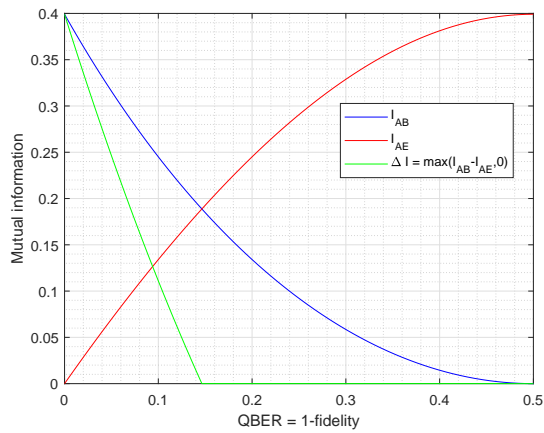
At first sight, these graphs may seem identical. However, we can see that the ordinates at the origin are different. For BB84 and the analogue of the no basis-switching protocol, its value is equal to 0,5 while for the two others it is 0,4. Moreover, if we superimpose the curves, we can see that the ones for BB84 and the discrete no basis-switching protocol are exactly the same and the two others are also equal. This has a consequence on the bit rate; they will be smaller for the analogue of Grosshans-Grangier and the noise-tolerant protocol than for BB84 and the discrete no basis-switching. It also appears that the QBER for which the two mutual information are equal is the same for all the protocols.



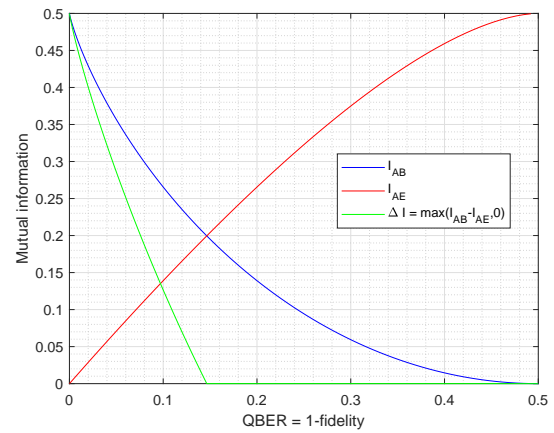
BB84.



Discrete noise-tolerant protocol.



Discrete Grosshans-Grangier protocol.



Discrete no basis-switching protocol.

Figure 6.8 – Representation of the mutual information and the bit rate for the four protocols in direct reconciliation.

6.5 Conclusion

In this chapter, we developed three new protocols and BB84. First, we have described all the stages of the protocol. Then these stages were formulated in term of states, POVM, cloning machine... in order to calculate the joint probability of the measurement for Bob and Eve conditional to the input of Alice. With this probability, it was possible to calculate the mutual information in order to determine if the new protocols can reach a higher security bound or a higher bit rate than in BB84. By doing the calculations, we can see that none of the new protocols presents any advantage with respect to BB84 in the case of direct reconciliation. However, the discrete no basis-switching protocol did show exactly the same curves and thus characteristics as BB84.

Chapter 7

Security of the four canonical protocols supplemented with reverse reconciliation

In the previous chapter, we studied four protocols for which we defined the states sent by Alice, the measurements done by Bob and Eve but also the action of Eve on the quantum channel. However, everything was studied in the case of direct reconciliation; the case where the final key is built from the bits possessed by Alice.

Now, we will analyse the other case: the reverse reconciliation. This means that the key will be built from the bits obtained by Bob. This will mainly change the attack of Eve. Indeed, before, she was trying to obtain data about the state of Alice. Now, she would like to possess data about the state of Bob. To do that, she will use another cloning machine that must be defined. The measurements done in the different protocols will not change (they will still be described by POVMs whose elements will depend on angles).

In order to study the protocols with reverse reconciliation, this chapter will be divided in two main parts: in the first one, we present the cloning machine that will be used by Eve while in the second, we trace the same kind of graphs as for direct reconciliation in order to compare the efficiency.

7.1 Cloning machine and protocol

7.1.1 Study of the cloning machine used by Eve

Eve wants to have a state which is correlated to the one of Bob. Her objective is then to get entangled with Bob. To obtain this, an idea is to use the reverse of the cloning machine used in the case of direct reconciliation. Indeed, we remember the two extreme cases of it: if the cloning angle was 0° , Bob did not receive any information and Eve did receive good information about the state of Alice. In the other extreme case (90°), it was the reverse. Such situations are presented in the figure 7.1 with the equation of the cloning machine corresponding to the two cases.

In the case of reverse reconciliation, these two extreme cases must change as Eve wants to be linked to Bob and not to Alice. These new desired extreme cases are presented in figure 7.2.

It appears that the two cases look similar but in the reverse way. One can then imagine a protocol in which Alice and Eve send an EPR state to the cloning machine. The cloning machine will then send one of the two states to Bob or a mix of them as a function of the cloning angle. This is the same idea as the one presented in the continuous case for

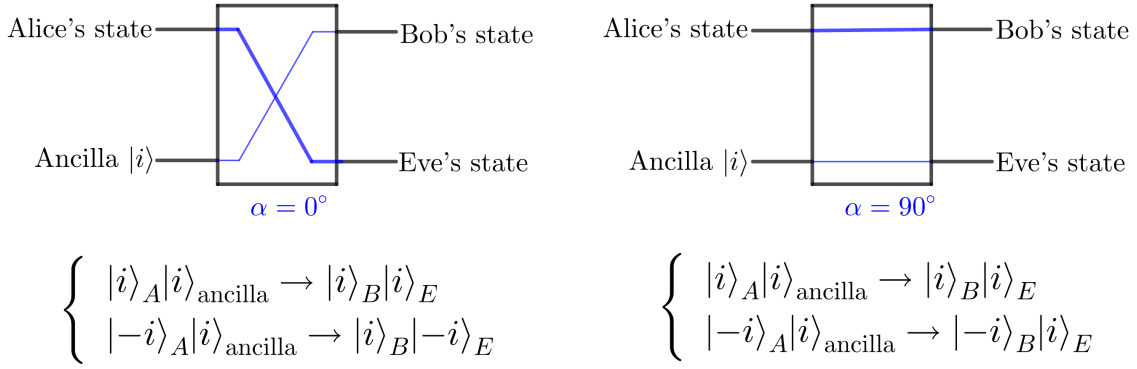


Figure 7.1 – Extreme situations of the cloning machine like it was used for direct reconciliation.

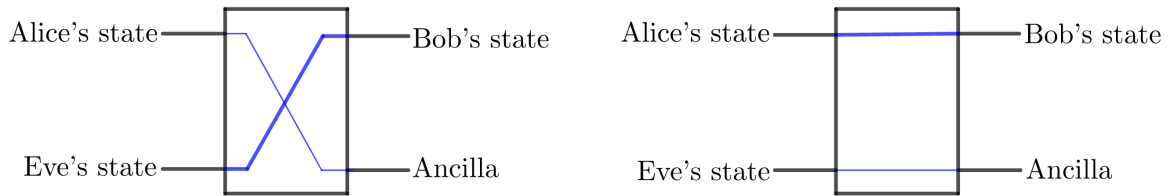


Figure 7.2 – Extreme situations of the cloning machine wanted for the reverse reconciliation.

the analysis of security. Indeed, as mentioned in chapter 3, with continuous variables, lots of analysis are done by imagining that Alice sends an EPR state and then makes a measurement on it. This is strictly equivalent to the case where she prepares a state that she will then send to Bob. These two visions are totally equivalent due to the entanglement. The scheme of this idea to make a protocol is presented in the figure 7.3. In this figure, we can see the action of the three actors:

- Alice measures one of the states of an EPR pair. This is exactly equivalent to what was done in direct reconciliation where she prepared a state and sent it to Bob. Indeed, if the EPR pair is the Bell's state $|\Phi_+\rangle$, when Alice will perform her measurement on the first qubit of this state, she will then know in which state the second qubit (the one sent to Bob) is. This is the same as if she prepares a state that she sends to Bob.
- Bob performs a measurement.
- Eve measures one of the qubit of an EPR pair and she applies the cloning machine. The second qubit that comes out of the cloning machine is thrown away in this idea of a protocol. However, we will see that throwing away this qubit is not a good idea. If the fourth qubit is thrown away the cloning machine will not do exactly what we want. The scheme of the protocol will be a little bit modified for that.

We see that the actions of Alice and Bob did not change with respect to direct reconciliation. There is just the action of Eve that changes as she wants information on Bob's state and no more on Alice's state.

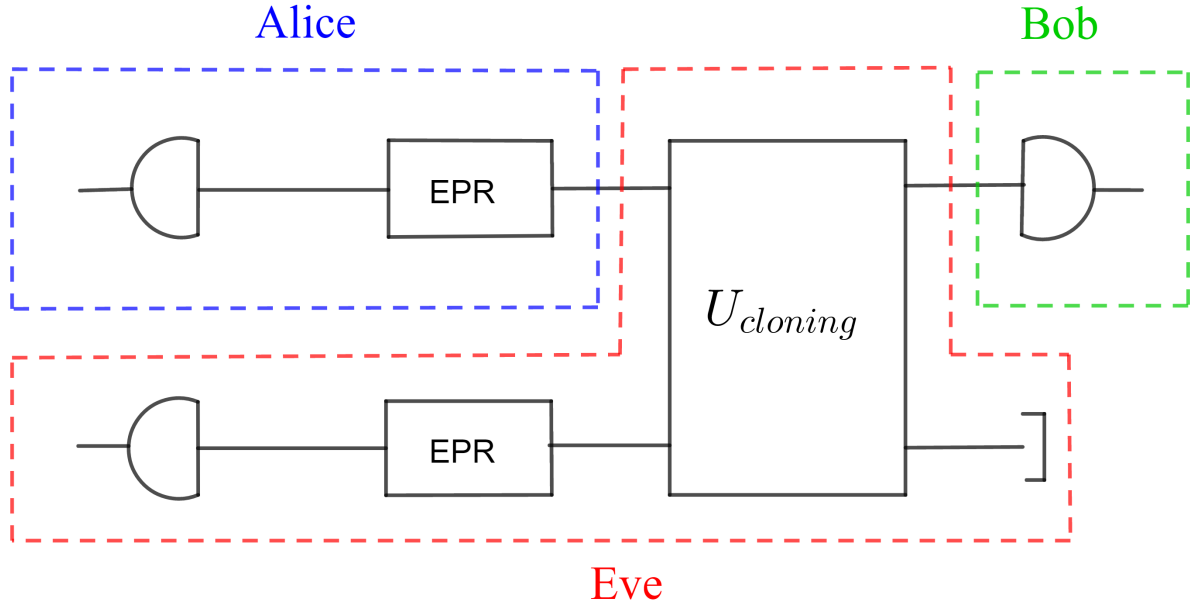


Figure 7.3 – Idea of a protocol for reverse reconciliation. The elements named EPR generates a pair of entangled states. Each time one state goes to the left while the second goes to the right. It will be seen in the following that this scheme does not exactly corresponds to what we want and that it must be modified.

In order to describe mathematically the different protocols, we need to know the action of the cloning machine which corresponds to a unitary operator. As mentioned before, the cloning machine must do the reverse operation compared to the previous one. If we have the representation of the operator under a matrix form, we can take the inverse of it and it will give us the unitary matrix that we need to describe the reverse reconciliation protocol. As the operator is unitary, its adjoint is equal to its inverse.

In order to find the matrix corresponding to the cloning machine, we will use the information that we had about the phase-covariant cloning machine. However, in the definition of the cloning machine that is usually given, there is only the transformation for two different input states. Indeed, we can recall this definition:

$$\begin{cases} |i\rangle |i\rangle \rightarrow |i\rangle |i\rangle \\ | -i\rangle |i\rangle \rightarrow \cos(\alpha) |i\rangle | -i\rangle + \sin(\alpha) | -i\rangle |i\rangle \end{cases} \quad (7.1)$$

The second input qubit is always $|i\rangle$ as it is an ancilla that is injected in the cloning machine because two qubits are needed at the output. However, if the second bit which is injected is different than $|i\rangle$, it is necessary to define the output states.

This can be done by trying to have a symmetric effect on the qubits. In equation 7.1, if the first bit is equal to $| -i\rangle$ while the second one is equal to $|i\rangle$, we obtain a combination of $|i\rangle | -i\rangle$ and $| -i\rangle |i\rangle$ at the output where the coefficients are given by the cosine and the sine of the angle of the cloning machine.

If we reverse the two input bits, we want something which is similar. This leads to:

$$|i\rangle | -i\rangle \rightarrow -\cos(\alpha) | -i\rangle |i\rangle + \sin(\alpha) |i\rangle | -i\rangle \quad (7.2)$$

The "-" is there because then, we find something which has the form of a rotation and it will be useful to obtain a unitary matrix. However, the sign "-" will disturb the calculation as the sign before the cosine should have been different in order to have

something which is totally symmetric. Finally, for the two input qubits equal to $|-i\rangle$, we must complete the matrix so that it is unitary:

$$U.U^\dagger = \mathbb{1} \quad (7.3)$$

The matrix of the cloning operation used in the direct reconciliation scheme is then:

$$U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & -\cos(\alpha) & \sin(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (7.4)$$

To find the cloning machine for the reverse reconciliation, we take the adjoint:

$$U_{cloning} = U^\dagger = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \sin(\alpha) & -\cos(\alpha) & 0 \\ 0 & \cos(\alpha) & \sin(\alpha) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (7.5)$$

This matrix correspond to the transformations:

$$\begin{cases} |i\rangle |i\rangle \rightarrow |i\rangle |i\rangle \\ |i\rangle |-i\rangle \rightarrow \cos(\alpha) |-i\rangle |i\rangle + \sin(\alpha) |i\rangle |-i\rangle \\ |-i\rangle |i\rangle \rightarrow -\cos(\alpha) |i\rangle |-i\rangle + \sin(\alpha) |-i\rangle |i\rangle \\ |-i\rangle |-i\rangle \rightarrow |-i\rangle |-i\rangle \end{cases} \quad (7.6)$$

7.1.2 Calculation of the joint probability of measurements

We can study the protocol. The various calculations were also done with Mathematica. The code can be found in annex A. Before studying the measurements, we must find the density matrix for the four qubits after the application of the cloning machine: the one kept by Alice from the first EPR pair, the one kept by Eve from the second EPR pair, the one sent to Bob after the cloning machine and finally the one that will be thrown away. To do so, the EPR states must be reformulated in the correct basis by using the equations found in the previous chapter (6.5).

The input state before the cloning machine is thus:

$$\left(\frac{|0\rangle |0\rangle + |1\rangle |1\rangle}{\sqrt{2}} \right)_{AA'} \left(\frac{|0\rangle |0\rangle + |1\rangle |1\rangle}{\sqrt{2}} \right)_{EE'} = \left(\frac{|i\rangle |-i\rangle + |-i\rangle |i\rangle}{\sqrt{2}} \right)_{AA'} \left(\frac{|i\rangle |-i\rangle + |-i\rangle |i\rangle}{\sqrt{2}} \right)_{EE'} \quad (7.7)$$

In this equation, A and A' correspond to the two qubits of the first EPR pair. The qubit A is the one kept by Alice while the A' is one of the two inputs of the cloner. For EE', it is exactly the same. By applying the unitary transformation corresponding to the cloning on the qubits A' and E', we obtain finally the density matrix describing the state of the four qubits:

$$\rho_{AEBW} \quad (7.8)$$

where W designates the qubit that we neglect. Before using it for the measurements, the spy must do something in order to take the minus sign into account.

The problem caused by this sign can be better understood when looking at the extreme case where the cloning angle is equal to 0° . We then rewrite the action of the cloning machine:

$$\begin{cases} |i\rangle |i\rangle \rightarrow |i\rangle |i\rangle \\ |i\rangle |-i\rangle \rightarrow |-i\rangle |i\rangle \\ |-i\rangle |i\rangle \rightarrow -|i\rangle |-i\rangle \\ |-i\rangle |-i\rangle \rightarrow |-i\rangle |-i\rangle \end{cases} \quad (7.9)$$

The state coming from the EPR pair of Eve (the second input) is indeed transferred to Bob but there is this minus sign. This is due to the fact that the state which is thrown away is not always in the state $|i\rangle$ like it should be. Indeed, we took the inverse of the cloning machine considered before. In its definition, there was only the effect on two states that was enunciated and for each of these states, the ancilla qubit was in the state $|i\rangle$. We have added to that the effect of the cloning machine on the two other possible inputs for which the ancilla qubit is in the state $|-i\rangle$. This causes the sign minus to appear. In order to modify the state to get rid of this minus sign, Eve has then to do a controlled sign flip as a function of the fourth state (the one that we neglected before and that was thrown away). If this state is in the state $|-i\rangle$, we have to do a sign flip on Eve's bit. This conditional sign flip will be done by calculating the operator and then applying it on our state:

$$\rho_{AEBW}^{modified} = \sigma_{z_{cond}} \rho_{AEBW} \sigma_{z_{cond}} \quad (7.10)$$

We know that a sign flip corresponds to the third Pauli's matrix:

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (7.11)$$

This leads to the definition of the operator that we must apply:

$$\sigma_{z_{cond}} = \mathbb{1}_A \otimes \sigma_{z_E} \otimes \mathbb{1}_B \otimes |-i\rangle \langle -i|_W + \mathbb{1}_A \otimes \mathbb{1}_E \otimes \mathbb{1}_B \otimes |-i\rangle \langle -i|_W \quad (7.12)$$

To do this transformation, the value of the fourth qubit that was neglected is important, the scheme of the protocol is thus different. The new one is presented in figure 7.4. After using the fourth qubit to make the sign flip, it becomes again useless and is traced out.

From the new density matrix obtained after the sign flip, we can find the one specific to Alice, Eve and Bob by doing a partial trace on the fourth qubit.

$$\rho_{AEB} = Tr_W(\rho_{AEBW}^{modified}) \quad (7.13)$$

With that, we can find the joint probability for Alice, Bob and Eve to measure certain angles exactly like we did for direct reconciliation:

$$p(\theta_a, \theta_e, \theta_b) = Tr(\rho_{AEB} \cdot M_{\theta_a} \otimes M_{\theta_e} \otimes M_{\theta_b}) \quad (7.14)$$

The operators M_{θ_i} are the elements of POVM used by Alice, Bob and Eve to perform their measurements. The index "i" stands for the different players: "a" for Alice, "b" for Bob and "e" for Eve. They are exactly identical to the ones that were used in the previous chapter:

$$\frac{c_i}{2} \begin{pmatrix} 1 & e^{-i\theta_i} \\ e^{i\theta_i} & 1 \end{pmatrix} \quad (7.15)$$

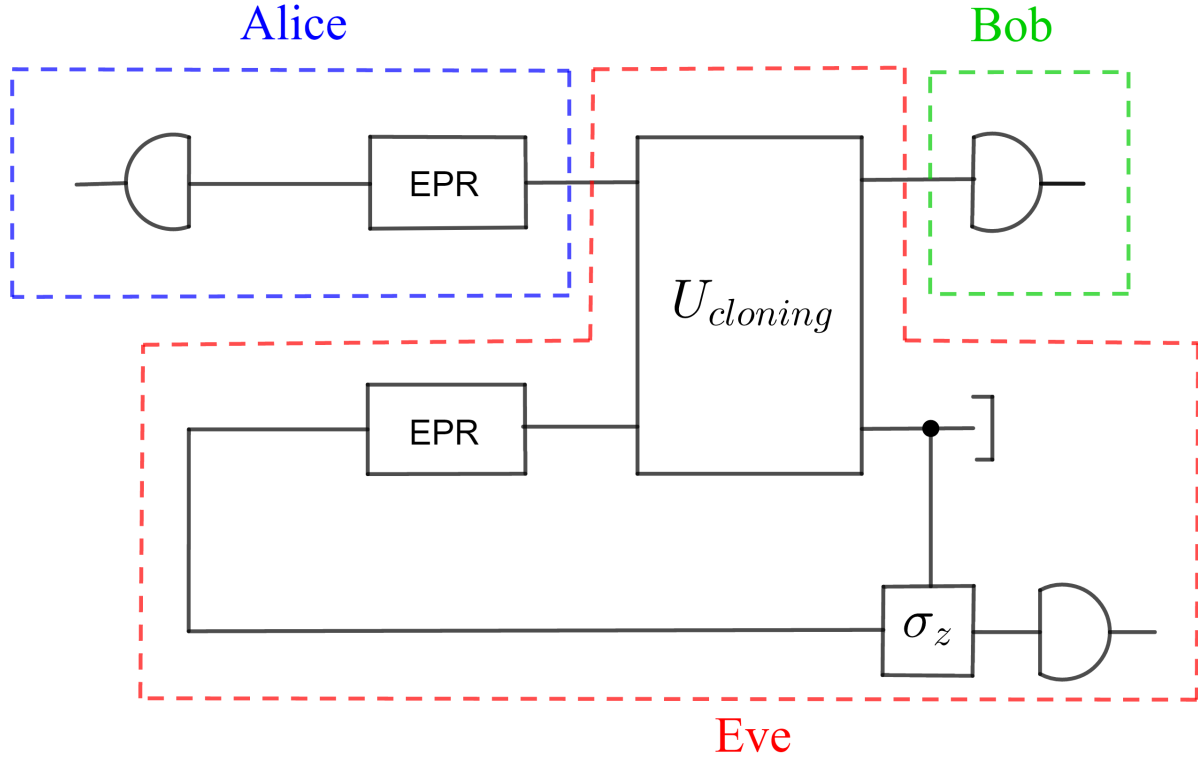


Figure 7.4 – Idea of a protocol for reverse reconciliation including the sign flip to correct the state of Eve.

The output of this calculation is:

$$p(\theta_a, \theta_e, \theta_b) = \frac{1}{8} c_{\theta_a} c_{\theta_b} c_{\theta_e} (\cos(\alpha) (\sin(\alpha) \cos(\theta_a - \theta_e) + \cos(\theta_b - \theta_e)) + \sin(\alpha) \cos(\theta_a - \theta_b) + 1) \quad (7.16)$$

With this joint probability, we can calculate all the probabilities and the entropies needed to compute the mutual information that will be used for the proof of security. One can already notice that this expression is quite similar to the expression found for the direct reconciliation scheme: the only differences are the coefficient before the expression and the fact that the angles of Alice and Bob are exchanged. The difference in the coefficient is due to the fact that here we have a joint probability while we had a conditional probability in the direct case. However, when replacing the values of the different constants c_i and multiplying by the probability for the state sent by Alice in the first case to obtain the joint probability, it appears that the two expressions become exactly identical except for the exchange of the angle of Alice and Bob.

7.1.3 Fidelity between Alice and Bob

Before being able to analyse properly the protocols in the reverse reconciliation scheme, we need to calculate the fidelity between Alice and Bob that will be used to calculate the QBER. The code to calculate the fidelity can be found in annex A.

To do that, we start by calculating the density matrix for Alice and Bob: $\rho_{AB} = Tr_E(\rho_{AEB})$. From this, we will obtain the fidelity using the equation 4.9 that gives the fidelity between a pure state and a density matrix. To calculate this fidelity, we will imagine that Bob has measured a certain pure state on the vertical plane considered until

now. This state can be formulated in the basis $\{|i\rangle; |-i\rangle\}$:

$$|\Psi_B\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} e^{-i\frac{\beta}{2}} \\ e^{i\frac{\beta}{2}} \end{pmatrix} \quad (7.17)$$

We can find the projector on this state: $\Pi_B = |\Psi_B\rangle \langle \Psi_B|$

It allows us to calculate the density matrix of Alice and Bob knowing that he has measured this state:

$$\rho_{AB|B} = \rho_{AB} \cdot (\mathbb{1}_A \otimes \Pi_B) \quad (7.18)$$

From this, the density matrix of A knowing the state of B can be found by making a partial trace and a normalization:

$$\rho_{A|B} = \frac{\text{Tr}_B(\rho_{AB|B})}{\text{Tr}(\rho_{AB|B})} \quad (7.19)$$

By writing the equation of the fidelity under another form, we will then be able to calculate it:

$$F(A, B) = \langle \Psi_B | \rho_{A|B} | \Psi_B \rangle \quad (7.20)$$

$$= \sum_n \langle \Psi_B | n \rangle \langle n | \rho_{A|B} | \Psi_B \rangle \quad (7.21)$$

$$= \sum_n \langle n | \rho_{A|B} | \Psi_B \rangle \langle \Psi_B | n \rangle \quad (7.22)$$

$$= \text{Tr}(\rho_{A|B} | \Psi_B \rangle \langle \Psi_B |) = \text{Tr}(\rho_{A|B} \cdot \Pi_B) \quad (7.23)$$

$$(7.24)$$

By making this calculation, we obtain the fidelity $\frac{1+\sin(\alpha)}{2}$.

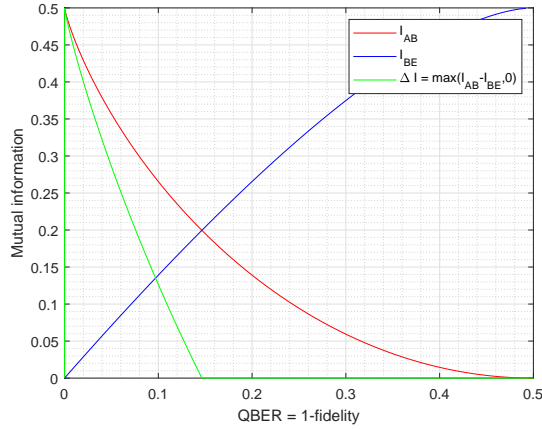
The fidelity is independent of the state measured by Bob which is coherent with the fact that we used an inverted phase covariant cloner. Once again, the result is the same as the one with direct reconciliation. With the fidelity and the joint probability, we have all the elements needed to trace the mutual information and to study the four protocols in order to compare them with the direct reconciliation protocols.

7.2 Analysis

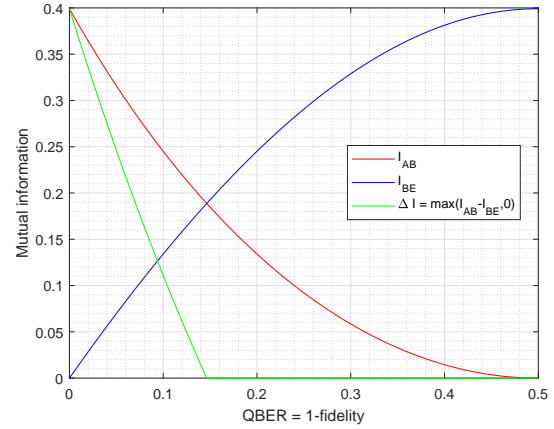
Like for direct reconciliation, the mutual information was obtained with Matlab by calculating the different probabilities and entropies by taking into account different values for the different protocols. The code can be found in annex A. The values varying from one protocol to the other are the constant before the elements of the POVM c_{θ_i} and also the angles of the elements. For these angles, the same values as for direct reconciliation were taken. The results are presented in the figure 7.5.

These graphs are exactly the same as the ones obtained for direct reconciliation. This can be explained because as it was mentioned before, the expression of the probabilities for direct reconciliation 6.15 and for reverse reconciliation 7.16 are identical if we exchange the angles of Alice and Bob which is logical when we observe the protocol: the roles of Alice and Bob are exchanged. This explains why the curves are exactly the same.

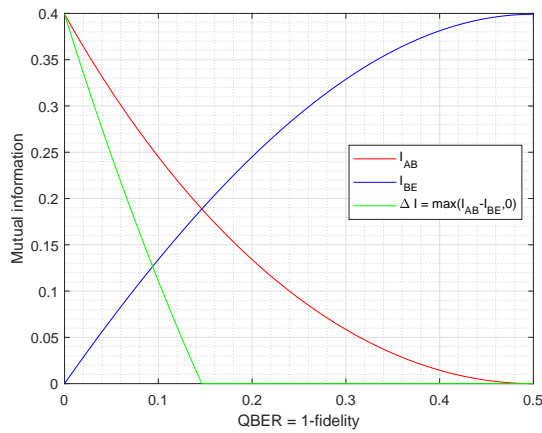
However, this result is quite unexpected. Indeed, the idea to analyse the different protocols with direct and reverse reconciliation came from the continuous variables. In



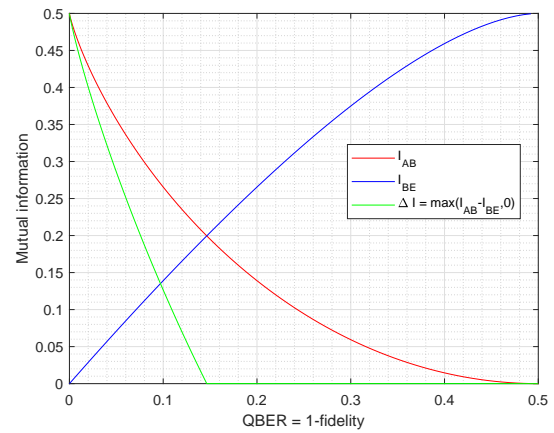
BB84.



Discrete noise-tolerant protocol.



Discrete Grosshans-Grangier protocol.



Discrete no basis-switching protocol.

Figure 7.5 – Representation of the mutual information and the bit rate for the four protocols in reverse reconciliation.

these protocols, there was an improvement in reverse reconciliation. The intuition was that such improvements would also be found for discrete variables. Indeed, the work done in the two last chapters was to look for the analogue version of the continuous protocol. We thus expected that the results would follow the same improvements as in continuous variables protocols. In the next chapter, we will try to find some tracks that could explain the difference between discrete and continuous quantum key distribution.

7.3 Conclusion

In this chapter, we searched for the analogue of the protocols with continuous variables that used reverse reconciliation. In order to define the protocols, we had to study the cloning machine that Eve would use to obtain information on Bob's state rather than on Alice's state. Defining this machine required the application of a controlled sign-flip in order to have the expected result for the cloning machine. Thereafter, the probability of measuring different angles could be found easily from a calculation similar to the one of last chapter. After calculating this probability, we determined the fidelity between Alice and Bob. With all these elements, we could analyse the different protocols, that turned out not to give improvements with respect to the direct reconciliation.

Chapter 8

Discussion and conclusion

The objective of this research was to try to find new protocols with discrete variables that could maybe present better properties than BB84. The two main properties that were observed are the QBER above which it is impossible to obtain a secure communication and the bit rate that can be obtained for the different QBER. The intuition was that better results would be obtained as is the case for continuous variables if reverse reconciliation is used.

This report was divided in two parts. The first part was focused on the definition of different tools as well as the presentation of the state of the art about quantum key distribution. The first chapter was dedicated to the fundamental notions of quantum mechanics with discrete variables among which the density matrix, the POVM,... Thereafter, the second chapter dealt with the Shannon information theory. In the framework of this thesis, this theory is very important as when the different measurements were performed, the end of the protocol became classical. To be sure that Alice and Bob could exchange a secret message, we had to calculate the mutual information which is a central element of Shannon's theory. The third chapter focused on cryptography protocols and a special part was dedicated to BB84 as it is one of the protocols that would be described in the second part of the thesis. The last chapter was about cloning machines. We presented the no-cloning theorem which states that any quantum state cannot be cloned perfectly. Even if it is impossible to clone perfectly a state, it is possible to make imperfect cloning. These imperfect cloning machines were also presented in that chapter.

The second part presented the new contributions in this thesis. It was divided in two main chapters. In the first one, the objective was to define the protocols. This has led us to the definition of POVM but also to a change of basis in the cloning machine in order to fit with the wanted scheme. Thereafter, the probabilities of different measurements were computed with the objective to calculate the security bounds. Once these were found, the coherence of the calculations was verified by comparing the results to the ones already known for BB84. Thereafter, the results were studied for the three new protocols but none has shown better properties than BB84. Finally, the reverse reconciliation was studied in the last chapter. Here again, no improvement with respect with direct reconciliation was observed, in contrast with our expectation.

Let us conclude by discussing the possible origin for these discrepancies between discrete-variable and continuous-variable protocols, especially the fact that reverse reconciliation does not give better results than direct reconciliation. We can see three main differences between discrete- and continuous-variable protocols that could explain our unexpected results, but a more detailed study should be done in order to confirm the mechanism.

The first difference is related to the different status of noise and losses. Indeed, in

continuous-variables protocols, noise is always present. If a squeezed or coherent state is sent by Alice, this state always exhibits quantum noise on its two quadratures at Bob's side, even if everything was perfect. In contrast, with a discrete-variable protocol, there is no noise on the detector's side when the line is perfect. The line losses simply appear as added noise in discrete-variable protocols (due to the dark counts of the detectors), but otherwise they decrease the key rate without giving an advantage to Bob. In contrast, in continuous-variable protocols, line losses have a very detrimental effect as the fraction of the input beam that does not reach Bob could potentially be exploited by Eve. This is why in direct reconciliation, the maximum tolerable loss is only 3 dB in continuous-variable protocols. Reverse reconciliation was specifically devised to allow for higher losses in such protocols. We believe that a way of finding the same advantage with discrete-variable protocols could be to consider that Bob measures a ternary observable (where, in addition to 0 and 1, the third possible outcome would correspond to when the detector does not click). If this was taken into account in our analysis of discrete-variable protocols, reverse reconciliation might potentially lead to some advantage.

A second difference between discrete- and continuous-variable protocols is connected with the measurement step, in particular with heterodyne measurements in which the two quadratures are measured simultaneously. In our analogous discrete version, heterodyne measurement is described by a POVM measurement with four elements (associated with two output bits). However, given its structure, such a POVM happens to be realizable as a random choice between two bases followed by a projective measurement discriminating the two states of the chosen basis. This is qualitatively very different from a heterodyne measurement, which cannot be realized as a random basis choice followed by homodyne measurement.

A third difference is related to the fact that there is no maximally-entangled state for continuous variables (this would imply a divergence of the mean photon number). Thus, in continuous-variable protocols, the squeezed states sent by Alice over the channel to Bob always have a finite squeezing. This means that their variance is not equal to zero. Even in the entanglement-based equivalent protocol, Alice's homodyne measurement (built on a resolution of the identity with infinitely squeezed states) gives rise to finite squeezed states that are sent over the channel to Bob. However, with his homodyne measurement, Bob projects onto an infinitely squeezed states (built on a resolution of the identity). Therefore, when reversing the time evolution as we do in reverse reconciliation, the states that are back-propagated (i.e., Bob's projected states) have infinite squeezing, while the states sent by Alice in direct reconciliation have finite squeezing. Such a difference between direct and reverse does not exist in discrete-variable protocols, which is perhaps why we did not observe any difference between direct and reverse reconciliation.

As a conclusion, we have established that all the discrete-variable protocols that we have studied have the same efficiency as BB84, or a even lower one. In this sense, BB84 should then be viewed as the best protocol in this family. This conclusion should, however, be tempered by that it is conditional on the specific model we have studied and on the hypotheses we have made (e.g. restricting to cloning-based individual attacks). Notably, if losses had been incorporated in our discrete-variable protocols, it remains possible that reverse reconciliation may prove useful as with continuous-variable protocols.

Bibliography

- [1] Charles H. Bennett. “Quantum cryptography using any two nonorthogonal states”. en. In: *Physical Review Letters* 68.21 (May 1992), pp. 3121–3124. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.68.3121.
- [2] Charles H. Bennett and Gilles Brassard. “Quantum cryptography: public key distribution and coin tossing”. In: *Theoretical Computer Science* 560 (1984), pp. 7–11.
- [3] D. Bruss et al. “Phase covariant quantum cloning”. en. In: *Physical Review A* 62.1 (June 2000). arXiv: quant-ph/9909046. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.62.012302.
- [4] V. Bužek and M. Hillery. “Quantum copying: Beyond the no-cloning theorem”. en. In: *Physical Review A* 54.3 (Sept. 1996), pp. 1844–1852. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.54.1844.
- [5] Cerf. “Pauli cloning of a quantum Bit”. eng. In: *Physical review letters* 84.19 (2000), pp. 4497–4500. ISSN: 1079-7114.
- [6] N. J. Cerf, M. Lévy, and G. Van Assche. “Quantum distribution of Gaussian keys using squeezed states”. en. In: *Physical Review A* 63.5 (Apr. 2001). ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.63.052311.
- [7] Nicolas J. Cerf and Jaromír Fiurášek. “Optical quantum cloning”. en. In: *Progress in Optics*. Vol. 49. Elsevier, 2006, pp. 455–545. ISBN: 978-0-444-52732-5. DOI: 10.1016/S0079-6638(06)49006-5.
- [8] Nicolas J. Cerf and Philippe Grangier. “From quantum cloning to quantum key distribution with continuous variables: a review (Invited)”. en. In: *Journal of the Optical Society of America B* 24.2 (Feb. 2007), p. 324. ISSN: 0740-3224, 1520-8540. DOI: 10.1364/JOSAB.24.000324.
- [9] P. T. Cochrane, T. C. Ralph, and A. Dolińska. “Optimal cloning for finite distributions of coherent states”. en. In: *Physical Review A* 69.4 (Apr. 2004), p. 042313. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.69.042313.
- [10] Thomas M Cover and Joy A Thomas. *Elements on information theory*. en. 2006, p. 774.
- [11] Imre Csiszár and János Körner. “Broadcast Channels with Confidential”. en. In: *IEEE Transactions on information theory* IT-24.3 (May 1978), p. 10.
- [12] D. Dieks. “Communication by EPR devices”. en. In: *Physics Letters A* 92.6 (Nov. 1982), pp. 271–272. ISSN: 03759601. DOI: 10.1016/0375-9601(82)90084-6.
- [13] A.R. Dixon et al. “Gigahertz decoy quantum key distribution with 1 Mbit/s secure key rate”. In: *Optics Express* 16.23 (2008), pp. 18790–18791. ISSN: 10944087.

- [14] A. Einstein, B. Podolsky, and N. Rosen. “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete? (Book review)”. eng. In: *Physical Review* 47.10 (1935), pp. 777–780. ISSN: 0031-899X.
- [15] Artur K. Ekert. “Quantum cryptography based on Bell’s theorem”. en. In: *Physical Review Letters* 67.6 (Aug. 1991), pp. 661–663. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.67.661.
- [16] Jaromír Fiurášek and Nicolas J. Cerf. “Quantum cloning of a pair of orthogonally polarized photons with linear optics”. en. In: *Physical Review A* 77.5 (May 2008). ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.77.052308.
- [17] Raúl García-Patrón and Nicolas J. Cerf. “Continuous-Variable Quantum Key Distribution Protocols Over Noisy Channels”. en. In: *Physical Review Letters* 102.13 (Mar. 2009). ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.102.130501.
- [18] Nicolas Gisin et al. “Quantum cryptography”. In: *Reviews of Modern Physics* 74.1 (2002). ISSN: 0034-6861.
- [19] Daniel Gottesman and John Preskill. “Secure quantum key distribution using squeezed states”. en. In: *Physical Review A* 63.2 (Jan. 2001), p. 022309. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.63.022309.
- [20] Frédéric Grosshans and Nicolas J. Cerf. “Continuous-Variable Quantum Cryptography is Secure against Non-Gaussian Attacks”. en. In: *Physical Review Letters* 92.4 (Jan. 2004). ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.92.047905.
- [21] Frédéric Grosshans and Philippe Grangier. “Continuous Variable Quantum Cryptography Using Coherent States”. en. In: *Physical Review Letters* 88.5 (Jan. 2002), p. 057902. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.88.057902.
- [22] Frédéric Grosshans et al. “Quantum key distribution using gaussian-modulated coherent states”. In: *Nature* 421.6920 (2003). ISSN: 0028-0836.
- [23] Frédéric Grosshans et al. “Virtual entanglement and reconciliation protocols for quantum cryptography with continuous variables”. In: *Quantum Information and Computation* 3.Special (2003), pp. 535–552.
- [24] Nick Herbert. “FLASH: A superluminal communicator based upon a new kind of quantum measurement”. en. In: *Foundations of Physics* 12.12 (Dec. 1982), pp. 1171–1179. ISSN: 0015-9018, 1572-9516. DOI: 10.1007/BF00729622.
- [25] S. Lorenz, N. Korolkova, and G. Leuchs. “Continuous-variable quantum key distribution using polarization encoding and post selection”. en. In: *Applied Physics B* 79.3 (Aug. 2004), pp. 273–277. ISSN: 0946-2171, 1432-0649. DOI: 10.1007/s00340-004-1574-7.
- [26] Leonard Mandel. “Is a photon amplifier always polarization dependent?” In: *Nature* 304 (1983), p. 188.
- [27] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2010. ISBN: 978-1-107-00217-3.
- [28] R. L. Rivest, A. Shamir, and L. Adleman. *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*. eng. 1978.
- [29] Valerio Scarani et al. “Quantum cryptography protocols robust against photon number splitting attacks for weak laser pulses implementations”. en. In: *Physical Review Letters* 92.5 (Feb. 2004). arXiv: quant-ph/0211131, p. 057901. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.92.057901.

- [30] Claude Shannon and Warren Weaver. *The Mathematical Theory of Communication*. en. 1964, p. 131.
- [31] Peter W. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. en. In: *SIAM Journal on Computing* 26.5 (Oct. 1997). arXiv: quant-ph/9508027, pp. 1484–1509. ISSN: 0097-5397, 1095-7111. DOI: 10.1137/S0097539795293172.
- [32] G. S. Vernam. “Cipher printing telegraph systems for secret wire and radio telegraphic communication”. In: *Journal of the American Institute of Electrical Engineers* 45 (1926), p. 109.
- [33] Christian Weedbrook et al. “Quantum Cryptography Without Switching”. en. In: *Physical Review Letters* 93.17 (Oct. 2004), p. 170504. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.93.170504.
- [34] Colin P. Williams. *Explorations in quantum computing*. en. 2nd ed. Texts in computer science. OCLC: ocn700397520. London ; New York: Springer, 2011. ISBN: 978-1-84628-886-9.
- [35] W. K. Wootters and W. H. Zurek. “A single quantum cannot be cloned”. en. In: *Nature* 299.5886 (Oct. 1982), pp. 802–803. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/299802a0.

Appendix A

Codes for the numerical simulations

In this annex, all the codes that were used in order to obtain the graphs are presented. The annex is divided in two sections and each section correspond to the analysis of the protocols with either direct or reverse reconciliation.

A.1 Protocols with direct reconciliation

The codes presented in this section are the ones that were used to make the different calculations of the chapter 6.

A.1.1 Calculation of the probability with Mathematica

The first step of these calculations was the calculation of the probability for Bob and Eve to measure certain angles. The calculations were done using Mathematica:

In[1]:= (* Density matrix for Bob and Eve *)

In[2]:= **m** =

$$\frac{1}{2} \begin{pmatrix} \begin{array}{|c|c|c|c|} \hline 1 & \cos[\alpha] \exp[-i \theta] & \sin[\alpha] \exp[-i \theta] & 0 \\ \hline \cos[\alpha] \exp[i \theta] & (\cos[\alpha])^2 & \cos[\alpha] \sin[\alpha] & 0 \\ \hline \sin[\alpha] \exp[i \theta] & \cos[\alpha] \sin[\alpha] & (\sin[\alpha])^2 & 0 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array} \end{pmatrix}$$

$$\text{Out[2]} = \left\{ \left\{ \frac{1}{2}, \frac{1}{2} e^{-i \theta} \cos[\alpha], \frac{1}{2} e^{-i \theta} \sin[\alpha], 0 \right\}, \right. \\ \left\{ \frac{1}{2} e^{i \theta} \cos[\alpha], \frac{\cos[\alpha]^2}{2}, \frac{1}{2} \cos[\alpha] \sin[\alpha], 0 \right\}, \\ \left. \left\{ \frac{1}{2} e^{i \theta} \sin[\alpha], \frac{1}{2} \cos[\alpha] \sin[\alpha], \frac{\sin[\alpha]^2}{2}, 0 \right\}, \{0, 0, 0, 0\} \right\}$$

In[3]:= (* Measurement operators *)

$$\text{In[4]}:= \mathbf{M}_b = \frac{c_b}{2} \begin{pmatrix} \begin{array}{|c|c|} \hline 1 & \exp[-i \theta_b] \\ \hline \exp[i \theta_b] & 1 \\ \hline \end{array} \end{pmatrix}$$

$$\text{Out[4]} = \left\{ \left\{ \frac{c_b}{2}, \frac{1}{2} c_b e^{-i \theta_b} \right\}, \left\{ \frac{1}{2} c_b e^{i \theta_b}, \frac{c_b}{2} \right\} \right\}$$

$$\text{In[5]}:= \mathbf{M}_e = \frac{c_e}{2} \begin{pmatrix} \begin{array}{|c|c|} \hline 1 & \exp[-i \theta_e] \\ \hline \exp[i \theta_e] & 1 \\ \hline \end{array} \end{pmatrix}$$

$$\text{Out[5]} = \left\{ \left\{ \frac{c_e}{2}, \frac{1}{2} c_e e^{-i \theta_e} \right\}, \left\{ \frac{1}{2} c_e e^{i \theta_e}, \frac{c_e}{2} \right\} \right\}$$

In[6]:= **Mtot** = KroneckerProduct[Mb, Me]

[produit Kronecker]

$$\text{Out[6]} = \left\{ \left\{ \frac{c_b c_e}{4}, \frac{1}{4} c_b c_e e^{-i \theta_e}, \frac{1}{4} c_b c_e e^{-i \theta_b}, \frac{1}{4} c_b c_e e^{-i \theta_b - i \theta_e} \right\}, \right. \\ \left\{ \frac{1}{4} c_b c_e e^{i \theta_e}, \frac{c_b c_e}{4}, \frac{1}{4} c_b c_e e^{-i \theta_b + i \theta_e}, \frac{1}{4} c_b c_e e^{-i \theta_b} \right\}, \\ \left\{ \frac{1}{4} c_b c_e e^{i \theta_b}, \frac{1}{4} c_b c_e e^{i \theta_b - i \theta_e}, \frac{c_b c_e}{4}, \frac{1}{4} c_b c_e e^{-i \theta_e} \right\}, \\ \left. \left\{ \frac{1}{4} c_b c_e e^{i \theta_b + i \theta_e}, \frac{1}{4} c_b c_e e^{i \theta_b}, \frac{1}{4} c_b c_e e^{i \theta_e}, \frac{c_b c_e}{4} \right\} \right\}$$

In[7]:= (* Calculation of the probability *)

In[8]:= **final = Simplify[m.Mtot]**
[simplifie]

$$\text{Out[8]} = \left\{ \left\{ \begin{aligned} &\frac{1}{8} \text{cb ce } e^{-i \theta} \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i (\theta + \theta_{\text{tae}})} \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i (\theta + \theta_{\text{tab}})} \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i (\theta + \theta_{\text{tab}} + \theta_{\text{tae}})} \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right) \end{aligned} \right\}, \right. \\ \left. \left\{ \begin{aligned} &\frac{1}{8} \text{cb ce } \cos[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i \theta_{\text{tae}}} \cos[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i \theta_{\text{tab}}} \cos[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i (\theta_{\text{tab}} + \theta_{\text{tae}})} \cos[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right) \end{aligned} \right\}, \right. \\ \left. \left\{ \begin{aligned} &\frac{1}{8} \text{cb ce } \sin[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i \theta_{\text{tae}}} \sin[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i \theta_{\text{tab}}} \sin[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right), \\ &\frac{1}{8} \text{cb ce } e^{-i (\theta_{\text{tab}} + \theta_{\text{tae}})} \sin[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right) \end{aligned} \right\}, \{0, 0, 0, 0\} \right\}$$

In[9]:= **probability = Tr[final]**
[trace tr]

$$\text{Out[9]} = \frac{1}{8} \text{cb ce } e^{-i \theta} \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right) + \\ \frac{1}{8} \text{cb ce } e^{-i \theta_{\text{tae}}} \cos[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right) + \\ \frac{1}{8} \text{cb ce } e^{-i \theta_{\text{tab}}} \sin[\alpha] \left(e^{i \theta} + e^{i \theta_{\text{tae}}} \cos[\alpha] + e^{i \theta_{\text{tab}}} \sin[\alpha] \right)$$

In[10]:= **FullSimplify[probability]**
[simplifie complètement]

$$\text{Out[10]} = \frac{1}{8} \text{cb ce } \left(1 + \cos[\alpha]^2 + 2 \cos[\theta - \theta_{\text{tae}}] \sin[\alpha] + \sin[\alpha]^2 + \right. \\ \left. 2 \cos[\alpha] \left(\cos[\theta - \theta_{\text{tae}}] + \cos[\theta_{\text{tab}} - \theta_{\text{tae}}] \sin[\alpha] \right) \right)$$

A.1.2 Analysis of the different protocols

Once the probability was obtained, the different graphs showing the mutual information were obtained using a code in Matlab. They were used to analyse the security of all the protocols and to compare them to BB84.

```

1  % Conditional probability for the measure of the angles theta_b
   and theta_e by Bob and Eve if Alice sent the angle theta_a
2
3  f = @(c_b,c_e,theta_a ,alpha_cloner ,theta_b ,theta_e) (1/4)*c_b*
   c_e*(1+cos(alpha_cloner).*cos(theta_a - theta_e)+cos(theta_a
   - theta_b).*sin(alpha_cloner)+cos(alpha_cloner).*cos(
   theta_b - theta_e).*sin(alpha_cloner));
4
5
6  % Selection of the protocol
7  BB84 = true;
8  noise = false;
9  australian = false;
10 GrosshansGrangier = false;
11
12 % Parameters depending on the protocol
13 if (noise)
14     AngleAlice = pi/2:pi:3*pi/2;
15     AngleBob = pi/4:pi/2:7*pi/4;
16     AngleEve = pi/4:pi/2:7*pi/4;
17     cBob = 1/2;
18     cEve = 1/2;
19     p_Alice = 1/2;
20 end
21 if (BB84)
22     AngleAlice = 0:pi:pi;
23     AngleBob = 0:pi:pi;
24     AngleEve = 0:pi:pi;
25     cBob = 1;
26     cEve = 1;
27     p_Alice = 1/2;
28 end
29 if (australian)
30     AngleAlice = pi/4:pi/2:7*pi/4;
31     AngleBob = pi/4:pi/2:7*pi/4;
32     AngleEve = pi/4:pi/2:7*pi/4;
33     cBob = 1/2;
34     cEve = 1/2;
35     p_Alice = 1/4;
36 end
37 if (GrosshansGrangier)
38     AngleAlice = pi/4:pi/2:7*pi/4;
39     AngleBob = pi/2:pi:3*pi/2;
40     AngleEve = pi/2:pi:3*pi/2;
41     cBob = 1;

```

```

42     cEve = 1;
43     p_Alice = 1/4;
44 end
45
46
47 % Joint probability
48
49 h = @(alpha, Alice, Bob, Eve) f(cBob, cEve, Alice, alpha, Bob, Eve);
50 jointproba = @(a, Alice, Bob, Eve) p_Alice * h(a, Alice, Bob, Eve);
51
52 % Calculation of the different probabilities
53 proba_a_and_b = @(alpha, Alice, Bob) 0;
54 for i = 1:length(AngleEve)
55     proba_a_and_b = @(alpha, Alice, Bob) (proba_a_and_b(alpha,
56         Alice, Bob) + jointproba(alpha, Alice, Bob, AngleEve(i)));
57 end
58 proba_a_and_e = @(alpha, Alice, Eve) 0;
59 for j = 1:length(AngleBob)
60     proba_a_and_e = @(alpha, Alice, Eve) (proba_a_and_e(alpha,
61         Alice, Eve) + jointproba(alpha, Alice, AngleBob(j), Eve));
62 end
63 proba_a = @(alpha, Alice) 0;
64 for l = 1:length(AngleBob)
65     proba_a = @(alpha, Alice) (proba_a(alpha, Alice) +
66         proba_a_and_b(alpha, Alice, AngleBob(l)));
67 end
68 proba_b = @(alpha, Bob) 0;
69 for m = 1:length(AngleAlice)
70     proba_b = @(alpha, Bob) (proba_b(alpha, Bob) + proba_a_and_b(
71         alpha, AngleAlice(m), Bob));
72 end
73 proba_e = @(alpha, Eve) 0;
74 for n = 1:length(AngleAlice)
75     proba_e = @(alpha, Eve) (proba_e(alpha, Eve) + proba_a_and_e(
76         alpha, AngleAlice(n), Eve));
77 end
78
79 % Calculation of the entropies
80 H_A = @(alpha) 0;
81 for o = 1:length(AngleAlice)
82     H_A = @(alpha) (H_A(alpha) - proba_a(alpha, AngleAlice(o)) .*
83         log2(proba_a(alpha, AngleAlice(o))));
84 end
85 H_B = @(alpha) 0;
86 for p = 1:length(AngleBob)
87     H_B = @(alpha) (H_B(alpha) - proba_b(alpha, AngleBob(p)) .* log2(
88         proba_b(alpha, AngleBob(p))));
89 end
90 H_E = @(alpha) 0;
91 for q = 1:length(AngleEve)

```

```

85     H_E = @(alpha) (H_E(alpha)-proba_e(alpha, AngleEve(q)).*log2(
        proba_e(alpha, AngleEve(q))));
86 end
87 H_AB = @(alpha) 0;
88 for r = 1:length(AngleBob)
89     for s = 1:length(AngleAlice)
90         H_AB = @(alpha) (H_AB(alpha)-proba_a_and_b(alpha,
            AngleAlice(s), AngleBob(r)).*log2(proba_a_and_b(alpha,
            AngleAlice(s), AngleBob(r))));
91     end
92 end
93 H_AE = @(alpha) 0;
94 for t = 1:length(AngleEve)
95     for u = 1:length(AngleAlice)
96         H_AE = @(alpha) (H_AE(alpha)-proba_a_and_e(alpha,
            AngleAlice(u), AngleEve(t)).*log2(proba_a_and_e(alpha,
            AngleAlice(u), AngleEve(t))));
97     end
98 end
99
100 % Calculation of the mutual information
101 I_AB = @(alpha) (H_A(alpha)+H_B(alpha)-H_AB(alpha));
102 I_AE = @(alpha) (H_A(alpha)+H_E(alpha)-H_AE(alpha));
103
104 % Plot
105 alpha = 0:(pi/200):pi/2;
106 QBER = 1-((1/2)*(1+sin(alpha)));
107 y = real(I_AB(alpha));
108 z = real(I_AE(alpha));
109 plot(QBER, y/2, 'b')
110 hold on;
111 plot(QBER, z/2, 'r')
112 grid on; grid minor;
113 plot(QBER, max(y-z, 0)/2, 'g')
114 axis([0 0.5 0 0.5])
115 xlabel('QBER = 1-fidelity')
116 ylabel('Mutual information')
117 legend('I_A_B', 'I_A_E', '\Delta I = max(I_A_B-I_A_E, 0)')
118
119
120 if (BB84)
121     figure(2)
122     % Theoretical values for BB84
123     test = 1-(-QBER.*(log2(QBER)) - (1-QBER).*(log2((1-QBER))));
124     xBB = (1 + sin(acos(-2.*QBER + 1)))/2;
125     IAE_BB = 1 + xBB.*log2(xBB) +(1-xBB).*log2(1-xBB);
126
127     % Plot of the difference simulation/theory
128     plot(QBER, test-y, 'b')
129     hold on; grid on;

```

```

130 plot(QBER,IAEBB-z,'r')
131 plot(QBER,zeros(1,length(QBER)),'g')
132 legend('Difference I_A_B with theoretical I_A_B','Difference
      I_A_E with theoretical I_A_E')
133
134 % Theoretical curves for BB84
135 figure(3)
136 plot(QBER,test,'b')
137 hold on;
138 plot(QBER,IAEBB,'r')
139 xlabel('QBER = 1-fidelity')
140 ylabel('Mutual information')
141 legend('I_A_B','I_A_E')
142 end

```

A.1.3 Evolution of the probability

Finally, the different probabilities were analysed with another code in Matlab.

```

1 clear all; close all;
2
3 % Conditional probability for the measure of Bob and Eve as a
  function of the state sent by Alice
4
5 f = @(c_b,c_e,theta_a,alpha,theta_b,theta_e) (1/4)*c_b*c_e*(1+
  cos(alpha).*cos(theta_a - theta_e)+cos(theta_a - theta_b).*
  sin(alpha)+cos(alpha).*cos(theta_b - theta_e).*sin(alpha));
6
7 % Parameters for BB84
8 AngleAlice = 0:pi:pi;
9 AngleBob = 0:pi:pi;
10 AngleEve = 0:pi:pi;
11 cBob = 1;
12 cEve = 1;
13 p_a = 1/2;
14
15 % Joint probability
16 h = @(alpha, Alice, Bob, Eve) f(cBob,cEve,Alice,alpha,Bob,Eve);
17 jointproba = @(a,Alice,Bob,Eve)p_a*h(a,Alice,Bob,Eve);
18
19
20 % Calculation of the different probabilities
21 proba_a_and_b = @(alpha,Alice,Bob) 0;
22 for i = 1:length(AngleEve)
23     proba_a_and_b = @(alpha,Alice,Bob) (proba_a_and_b(alpha,
      Alice,Bob)+jointproba(alpha,Alice,Bob, AngleEve(i)));
24 end
25 proba_a_and_e = @(alpha,Alice,Eve) 0;
26 for j = 1:length(AngleBob)
27     proba_a_and_e = @(alpha,Alice,Eve) (proba_a_and_e(alpha,
      Alice,Eve)+jointproba(alpha,Alice,AngleBob(j),Eve));

```

```

28 end
29
30 % State sent by Alice
31 Alice = 0;
32 Bob= 0:pi/200:2*pi;
33
34 % Graph for an angle of the cloning machine equal to 0
35 figure(1)
36 polarpattern(proba_a_and_b(0,Alice,Bob));
37 hold on;
38 polarpattern(proba_a_and_e(0,Alice,Bob));
39 legend('Bob','Eve')
40
41 % Graph for an angle of the cloning machine equal to pi/2
42 figure(2)
43 polarpattern(proba_a_and_b(pi/2,Alice,Bob),'EdgeColor','g');
44 hold on;
45 polarpattern(proba_a_and_e(pi/2,Alice,Bob));
46 legend('Bob','Eve')
47
48 % Graph showing the evolution of the probability as a function
   of the angle of the cloning machine
49 figure(3)
50 polarpattern(proba_a_and_b(0,Alice,Bob));
51 hold on;
52 polarpattern(proba_a_and_b(pi/10,Alice,Bob));
53 polarpattern(proba_a_and_b(pi/5,Alice,Bob));
54 polarpattern(proba_a_and_b(3*pi/10,Alice,Bob));
55 polarpattern(proba_a_and_b(pi/4,Alice,Bob));
56 polarpattern(proba_a_and_b(2*pi/5,Alice,Bob));
57 polarpattern(proba_a_and_b(pi/2,Alice,Bob));
58 legend('0','pi/10','pi/5','3*pi/10','pi/4','2*pi/5','pi/2')

```

A.2 Protocols with reverse reconciliation

The codes presented in this section are the ones that were used to make the different calculations of the chapter 7.

A.2.1 Calculation of the probability with Mathematica and of the fidelity between Alice and Bob

The first step of the calculation is to find the joint probability. In order to do that, it is necessary to inverse the matrix of the cloning machine and to apply the controlled sign flip on the bit of Eve. This is done in the following Mathematica code. Thereafter, the fidelity between Alice and Bob is also calculated.

In[1]:= (* Matrix of the cloning machine *)

$$\text{In[2]:= } U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \sin[\alpha] & -\cos[\alpha] & 0 \\ 0 & \cos[\alpha] & \sin[\alpha] & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Out[2]= {{1, 0, 0, 0}, {0, Sin[alpha], -Cos[alpha], 0},
{0, Cos[alpha], Sin[alpha], 0}, {0, 0, 0, 1}}

In[3]:= (* Adjoint of the matrix *)

In[4]:= Udag = Refine[ConjugateTranspose[U], alpha ∈ Reals]
 [affine] [transposé conjugué] [nombres]

Out[4]= {{1, 0, 0, 0}, {0, Sin[alpha], Cos[alpha], 0},
{0, -Cos[alpha], Sin[alpha], 0}, {0, 0, 0, 1}}

In[5]:= (* Verification that it is unitary *)

In[6]:= Verification = FullSimplify[U.Udag]
 [simplifie complètement]

Out[6]= {{1, 0, 0, 0}, {0, 1, 0, 0}, {0, 0, 1, 0}, {0, 0, 0, 1}}

In[7]:= (* Calculation of the state after the cloning machine *)

$$\text{In[8]:= } \text{Input1} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Output1 = U.Input1

Out[8]= {{0}, {0}, {0}, {1}}

Out[9]= {{0}, {0}, {0}, {1}}

$$\text{In[10]:= } \text{Input2} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

Output2 = U.Input2

Out[10]= {{0}, {0}, {1}, {0}}

Out[11]= {{0}, {-Cos[alpha]}, {Sin[alpha]}, {0}}

$$\text{In[12]:= } \text{Input3} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

Output3 = U.Input3

Out[12]= {{0}, {1}, {0}, {0}}

Out[13]= {{0}, {Sin[alpha]}, {Cos[alpha]}, {0}}

$$\text{In[14]:= Input4} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

Output4 = U.Input4

Out[14]= {{1}, {0}, {0}, {0}}

Out[15]= {{1}, {0}, {0}, {0}}

In[16]:= **one = KroneckerProduct[Input4, Output1]**

|produit Kronecker

Out[16]= {{0}, {0}, {0}, {1}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}}

In[17]:= **two = KroneckerProduct[Input3, Output2]**

|produit Kronecker

Out[17]= {{0}, {0}, {0}, {0}, {0}, {-Cos[alpha]},
{Sin[alpha]}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}}

In[18]:= **three = KroneckerProduct[Input2, Output3]**

|produit Kronecker

Out[18]= {{0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0},
{Sin[alpha]}, {Cos[alpha]}, {0}, {0}, {0}, {0}, {0}}

In[19]:= **four = KroneckerProduct[Input1, Output4]**

|produit Kronecker

Out[19]= {{0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {0}, {1}, {0}, {0}, {0}}

In[20]:= **FinalState = $\frac{1}{2}$ (one + two + three + four)**

Out[20]= {{0}, {0}, {0}, { $\frac{1}{2}$ }, {0}, {- $\frac{\text{Cos}[\text{alpha}]}{2}$ }, { $\frac{\text{Sin}[\text{alpha}]}{2}$ },
{0}, {0}, { $\frac{\text{Sin}[\text{alpha}]}{2}$ }, { $\frac{\text{Cos}[\text{alpha}]}{2}$ }, {0}, { $\frac{1}{2}$ }, {0}, {0}, {0}}

In[21]:= **BraFinal = Refine[ConjugateTranspose[FinalState], alpha ∈ Reals]**

|affine |transposé conjugué

|nombres

Out[21]= {{0, 0, 0, $\frac{1}{2}$, 0, - $\frac{\text{Cos}[\text{alpha}]}{2}$, $\frac{\text{Sin}[\text{alpha}]}{2}$, 0, 0, $\frac{\text{Sin}[\text{alpha}]}{2}$, $\frac{\text{Cos}[\text{alpha}]}{2}$, 0, $\frac{1}{2}$, 0, 0, 0}}

In[22]:= **(* Density matrix for the four bits *)**

In[23]:= **DensityMatrixFinal = FullSimplify[FinalState.BraFinal]**

[\[simplifie complètement\]](#)

Out[23]:=
$$\left\{ \begin{aligned} &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\left\{0, 0, 0, \frac{1}{4}, 0, -\frac{\cos[\alpha]}{4}, \frac{\sin[\alpha]}{4}, 0, 0, \frac{\sin[\alpha]}{4}, \frac{\cos[\alpha]}{4}, 0, \frac{1}{4}, 0, 0, 0\right\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\left\{0, 0, 0, -\frac{\cos[\alpha]}{4}, 0, \frac{\cos[\alpha]^2}{4}, -\frac{1}{4}\cos[\alpha]\sin[\alpha], 0, \right. \\ &\quad \left. 0, -\frac{1}{4}\cos[\alpha]\sin[\alpha], -\frac{1}{4}\cos[\alpha]^2, 0, -\frac{\cos[\alpha]}{4}, 0, 0, 0\right\}, \\ &\left\{0, 0, 0, \frac{\sin[\alpha]}{4}, 0, -\frac{1}{4}\cos[\alpha]\sin[\alpha], \frac{\sin[\alpha]^2}{4}, 0, \right. \\ &\quad \left. 0, \frac{\sin[\alpha]^2}{4}, \frac{1}{4}\cos[\alpha]\sin[\alpha], 0, \frac{\sin[\alpha]}{4}, 0, 0, 0\right\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\left\{0, 0, 0, \frac{\sin[\alpha]}{4}, 0, -\frac{1}{4}\cos[\alpha]\sin[\alpha], \frac{\sin[\alpha]^2}{4}, \right. \\ &\quad \left. 0, 0, \frac{\sin[\alpha]^2}{4}, \frac{1}{4}\cos[\alpha]\sin[\alpha], 0, \frac{\sin[\alpha]}{4}, 0, 0, 0\right\}, \\ &\left\{0, 0, 0, \frac{\cos[\alpha]}{4}, 0, -\frac{1}{4}\cos[\alpha]^2, \frac{1}{4}\cos[\alpha]\sin[\alpha], 0, \right. \\ &\quad \left. 0, \frac{1}{4}\cos[\alpha]\sin[\alpha], \frac{\cos[\alpha]^2}{4}, 0, \frac{\cos[\alpha]}{4}, 0, 0, 0\right\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\left\{0, 0, 0, \frac{1}{4}, 0, -\frac{\cos[\alpha]}{4}, \frac{\sin[\alpha]}{4}, 0, 0, \frac{\sin[\alpha]}{4}, \frac{\cos[\alpha]}{4}, 0, \frac{1}{4}, 0, 0, 0\right\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\} \end{aligned} \right\}$$

In[24]:= **(\star Function calculating different partial traces. Source: <http://>**

[\[fonction\]](#)

library.wolfram.com/infocenter/MathSource/5571/#downloads \star)

In[25]:= **SwapParts[expr_, pos1_, pos2_] :=**

ReplacePart[#, #, {pos1, pos2}, {pos2, pos1}] &[expr]

[\[remplace une partie\]](#)

TraceSystem[D_, s_] := (

Qubits = Reverse[Sort[s]];

[\[renverses\]](#) [\[trie\]](#)

TrkM = D;

[\[dérivée d\]](#)


```

z = (Dimensions[Qubits][[1]] + 1);
  [dimensions]

For[q = 1, q < z, q++,
  [pour chaque]
  n = Log[2, (Dimensions[TrkM][[1]])];
    [logarit... [dimensions]
  M = TrkM;
  k = Qubits[[q]];
  If[k == n,
    [si]
    TrkM = {};
    For[p = 1, p < 2n + 1, p = p + 2,
      [pour chaque]
      TrkM =
        Append[TrkM, Take[M[[p, All]], {1, 2n, 2}] + Take[M[[p + 1, All]], {2, 2n, 2}]];
        [appose [prends [tout [prends [tout]
    ],
  For[j = 0, j < (n - k), j++,
    [pour chaque]
    b = {};
    For[i = 1, i < 2n + 1, i++,
      [pour chaque]
      If[(Mod[(IntegerDigits[i - 1, 2, n][[n]] + IntegerDigits[i - 1, 2, n][[n - j - 1]]],
        [si [mod... [chiffres d'entier [chiffres d'entier]
        2]) == 1 && Count[b, i] == 0, Permut = {i, (FromDigits[
          [compte [depuis chiffres]
          SwapParts[(IntegerDigits[i - 1, 2, n]), {n}, {n - j - 1}], 2] + 1)}];
          [chiffres d'entier]
        b = Append[b, (FromDigits[SwapParts[(IntegerDigits[i - 1, 2, n]),
          [appose [depuis chiffres [chiffres d'entier]
          {n}, {n - j - 1}], 2] + 1)}];
        c = Range[2n];
        [plage]
        perm = SwapParts[c, {i}, {(FromDigits[
          [depuis chiffres]
          SwapParts[(IntegerDigits[i - 1, 2, n]), {n}, {n - j - 1}], 2] + 1)}];
          [chiffres d'entier]
        M = M[[perm, perm]];

    ]
  ]
];
TrkM = {};
For[p = 1, p < 2n + 1, p = p + 2,
  [pour chaque]
  TrkM =
    Append[TrkM, Take[M[[p, All]], {1, 2n, 2}] + Take[M[[p + 1, All]], {2, 2n, 2}]];
    [appose [prends [tout [prends [tout]
  ]
]

```

```

];
]

; Return[TrkM]
|reviens

```

In[27]:= (* Measurement operator *)

$$\text{In[28]:= } \mathbf{M}_a = \frac{c_a}{2} \left(\begin{array}{c|c} 1 & \text{Exp}[-i \theta_{taa}] \\ \hline \text{Exp}[i \theta_{taa}] & 1 \end{array} \right)$$

$$\text{Out[28]= } \left\{ \left\{ \frac{c_a}{2}, \frac{1}{2} c_a e^{-i \theta_{taa}} \right\}, \left\{ \frac{1}{2} c_a e^{i \theta_{taa}}, \frac{c_a}{2} \right\} \right\}$$

$$\text{In[29]:= } \mathbf{M}_b = \frac{c_b}{2} \left(\begin{array}{c|c} 1 & \text{Exp}[-i \theta_{tab}] \\ \hline \text{Exp}[i \theta_{tab}] & 1 \end{array} \right)$$

$$\text{Out[29]= } \left\{ \left\{ \frac{c_b}{2}, \frac{1}{2} c_b e^{-i \theta_{tab}} \right\}, \left\{ \frac{1}{2} c_b e^{i \theta_{tab}}, \frac{c_b}{2} \right\} \right\}$$

$$\text{In[30]:= } \mathbf{M}_c = \frac{c_e}{2} \left(\begin{array}{c|c} 1 & \text{Exp}[-i \theta_{tae}] \\ \hline \text{Exp}[i \theta_{tae}] & 1 \end{array} \right)$$

$$\text{Out[30]= } \left\{ \left\{ \frac{c_e}{2}, \frac{1}{2} c_e e^{-i \theta_{tae}} \right\}, \left\{ \frac{1}{2} c_e e^{i \theta_{tae}}, \frac{c_e}{2} \right\} \right\}$$

In[31]:= **Mtot = KroneckerProduct[Ma, Mc, Mb]**

[produit Kronecker]

$$\text{Out[31]} = \left\{ \left\{ \frac{ca\,cb\,ce}{8}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}-i\,\text{thetae}}, \right. \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}-i\,\text{thetab}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}-i\,\text{thetae}},$$

$$\left. \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}-i\,\text{thetab}-i\,\text{thetae}} \right\}, \left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}}, \frac{ca\,cb\,ce}{8}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}+i\,\text{thetab}},$$

$$\frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}+i\,\text{thetab}-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}-i\,\text{thetae}} \right\},$$

$$\left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}+i\,\text{thetae}}, \frac{ca\,cb\,ce}{8}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}-i\,\text{thetab}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}},$$

$$\left. \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}-i\,\text{thetab}} \right\}, \left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetae}}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}}, \frac{ca\,cb\,ce}{8}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}+i\,\text{thetab}+i\,\text{thetae}},$$

$$\frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}+i\,\text{thetab}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetaa}} \right\},$$

$$\left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}-i\,\text{thetab}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}-i\,\text{thetae}}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}-i\,\text{thetab}-i\,\text{thetae}}, \frac{ca\,cb\,ce}{8}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}},$$

$$\frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}-i\,\text{thetae}} \right\}, \left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}+i\,\text{thetab}}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}+i\,\text{thetab}-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}-i\,\text{thetae}},$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}}, \frac{ca\,cb\,ce}{8}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}-i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetae}} \right\},$$

$$\left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}-i\,\text{thetab}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}-i\,\text{thetab}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}+i\,\text{thetae}},$$

$$\frac{ca\,cb\,ce}{8}, \frac{1}{8}ca\,cb\,ce\,e^{-i\,\text{thetab}} \right\}, \left\{ \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}+i\,\text{thetab}+i\,\text{thetae}}, \right.$$

$$\frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}+i\,\text{thetab}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetaa}},$$

$$\left. \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}+i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetae}}, \frac{1}{8}ca\,cb\,ce\,e^{i\,\text{thetab}}, \frac{ca\,cb\,ce}{8} \right\}$$

In[32]:= **(* Application of the conditional sign-flip *)**

$$\text{In[33]:= identity} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\text{Out[33]= } \{ \{1, 0\}, \{0, 1\} \}$$

$$\text{In[34]:= signflip} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$\text{Out[34]= } \{ \{1, 0\}, \{0, -1\} \}$$

$$\text{In[35]:= u1} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\text{Out[35]= } \{ \{1, 0\}, \{0, 0\} \}$$

$$\text{In[36]:= u2} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\text{Out[36]= } \{ \{0, 0\}, \{0, 1\} \}$$

$$\begin{aligned} \text{In[37]:= total} &= \text{KroneckerProduct}[\text{identity}, \text{identity}, \text{identity}, \text{u1}] + \\ &\quad \text{KroneckerProduct}[\text{identity}, \text{signflip}, \text{identity}, \text{u2}] \end{aligned}$$

$$\begin{aligned} \text{Out[37]= } &\{ \{1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1, 0, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0\}, \\ &\{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -1\} \} \end{aligned}$$

```
In[38]:= matrixaftersignflip = total.DensityMatrixFinal.total
```

```
Out[38]= { {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0,  $\frac{1}{4}$ , 0,  $\frac{\cos[\alpha]}{4}$ ,  $\frac{\sin[\alpha]}{4}$ , 0, 0,  $\frac{\sin[\alpha]}{4}$ ,  $\frac{\cos[\alpha]}{4}$ , 0, 0,  $\frac{1}{4}$ , 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0,  $\frac{\cos[\alpha]}{4}$ , 0,  $\frac{\cos[\alpha]^2}{4}$ ,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ , 0,
  0,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ ,  $\frac{\cos[\alpha]^2}{4}$ , 0,  $\frac{\cos[\alpha]}{4}$ , 0, 0, 0},
  {0, 0, 0,  $\frac{\sin[\alpha]}{4}$ , 0,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ ,  $\frac{\sin[\alpha]^2}{4}$ , 0, 0,
   $\frac{\sin[\alpha]^2}{4}$ ,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ , 0,  $\frac{\sin[\alpha]}{4}$ , 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0,  $\frac{\sin[\alpha]}{4}$ , 0,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ ,  $\frac{\sin[\alpha]^2}{4}$ , 0,
  0,  $\frac{\sin[\alpha]^2}{4}$ ,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ , 0,  $\frac{\sin[\alpha]}{4}$ , 0, 0, 0},
  {0, 0, 0,  $\frac{\cos[\alpha]}{4}$ , 0,  $\frac{\cos[\alpha]^2}{4}$ ,  $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ , 0, 0,
   $\frac{1}{4}\cos[\alpha]\sin[\alpha]$ ,  $\frac{\cos[\alpha]^2}{4}$ , 0,  $\frac{\cos[\alpha]}{4}$ , 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0,  $\frac{1}{4}$ , 0,  $\frac{\cos[\alpha]}{4}$ ,  $\frac{\sin[\alpha]}{4}$ , 0, 0,  $\frac{\sin[\alpha]}{4}$ ,  $\frac{\cos[\alpha]}{4}$ , 0, 0,  $\frac{1}{4}$ , 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0},
  {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0} }
```

```
In[39]:= (* Tracing out of the fourth bit *)
```

```
In[40]:= rhoAEB = FullSimplify[TraceSystem[matrixaftersignflip, {4}]]
[simplifie complètement]
```

```
Out[40]= { {0, 0, 0, 0, 0, 0, 0, 0}, {0, 1/4, Cos[alpha]/4, Sin[alpha]/4, 0, 0, 0, 0},
  {0, Cos[alpha]/4, Cos[alpha]^2/4, 1/4 Cos[alpha] Sin[alpha], 0, 0, 0, 0},
  {0, 0, Sin[alpha]^2/4, 1/4 Cos[alpha] Sin[alpha], Sin[alpha]/4, 0, 0, 0},
  {0, Sin[alpha]/4, 1/4 Cos[alpha] Sin[alpha], Sin[alpha]^2/4, 0, 0, 0, 0},
  {0, 0, 1/4 Cos[alpha] Sin[alpha], Cos[alpha]^2/4, Cos[alpha]/4, 0, 0, 0},
  {0, 0, Sin[alpha]/4, Cos[alpha]/4, 1/4, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0} }
```

```
In[41]:= Proba = FullSimplify[Tr[rhoAEB.Mtot]]
[simplifie complètement] [trace tr]
```

```
Out[41]= 1/8 ca cb ce (1 + Cos[thetaa - thetab] Sin[alpha] +
  Cos[alpha] (Cos[thetab - thetae] + Cos[thetaa - thetae] Sin[alpha]))
```

```
In[42]:= (* Calculation of the density matrix
  of Alice and Bob used to calculate the fidelity *)
```

```
In[43]:= rhoAB = TraceSystem[rhoAEB, {2}]
```

```
Out[43]= { {Cos[alpha]^2/4, 0, 0, 0}, {0, 1/4 + Sin[alpha]^2/4, Sin[alpha]/2, 0},
  {0, Sin[alpha]/2, 1/4 + Sin[alpha]^2/4, 0}, {0, 0, 0, Cos[alpha]^2/4} }
```

In[44]:= (* Projector on the state of Bob *)

$$\text{In[45]:= } B = \frac{1}{\text{Sqrt}[2]} \begin{pmatrix} \cos\left[\frac{\text{angle}}{2}\right] - i \sin\left[\frac{\text{angle}}{2}\right] \\ \cos\left[\frac{\text{angle}}{2}\right] + i \sin\left[\frac{\text{angle}}{2}\right] \end{pmatrix}$$

$$\text{Out[45]= } \left\{ \left\{ \frac{\cos\left[\frac{\text{angle}}{2}\right] - i \sin\left[\frac{\text{angle}}{2}\right]}{\sqrt{2}} \right\}, \left\{ \frac{\cos\left[\frac{\text{angle}}{2}\right] + i \sin\left[\frac{\text{angle}}{2}\right]}{\sqrt{2}} \right\} \right\}$$

In[46]:= Bprim = Refine[ConjugateTranspose[B], angle ∈ Reals]
[affine [transposé conjugué [nombres

$$\text{Out[46]= } \left\{ \left\{ \frac{\cos\left[\frac{\text{angle}}{2}\right] + i \sin\left[\frac{\text{angle}}{2}\right]}{\sqrt{2}} \right\}, \left\{ \frac{\cos\left[\frac{\text{angle}}{2}\right] - i \sin\left[\frac{\text{angle}}{2}\right]}{\sqrt{2}} \right\} \right\}$$

In[47]:= projector = FullSimplify[B.Bprim]
[simplifie complètement

$$\text{Out[47]= } \left\{ \left\{ \frac{1}{2}, \frac{e^{-i \text{angle}}}{2} \right\}, \left\{ \frac{e^{i \text{angle}}}{2}, \frac{1}{2} \right\} \right\}$$

In[48]:= (* Identity on the state of A *)
[identité

$$\text{In[49]:= } \text{identityA} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

$$\text{Out[49]= } \{ \{1, 0\}, \{0, 1\} \}$$

In[50]:= totalprojector = KroneckerProduct[identityA, projector]
[produit Kronecker

$$\text{Out[50]= } \left\{ \left\{ \frac{1}{2}, \frac{e^{-i \text{angle}}}{2}, 0, 0 \right\}, \left\{ \frac{e^{i \text{angle}}}{2}, \frac{1}{2}, 0, 0 \right\}, \left\{ 0, 0, \frac{1}{2}, \frac{e^{-i \text{angle}}}{2} \right\}, \left\{ 0, 0, \frac{e^{i \text{angle}}}{2}, \frac{1}{2} \right\} \right\}$$

In[51]:= (* Calcul fidelity between Alice et Bob *)

In[52]:= projectedstate = FullSimplify[rhoAB.totalprojector]
[simplifie complètement

$$\begin{aligned} \text{Out[52]= } & \left\{ \left\{ \frac{\cos[\alpha]^2}{8}, \frac{1}{8} e^{-i \text{angle}} \cos[\alpha]^2, 0, 0 \right\}, \right. \\ & \left\{ \frac{1}{8} e^{i \text{angle}} (1 + \sin[\alpha]^2), \frac{1}{8} (1 + \sin[\alpha]^2), \frac{\sin[\alpha]}{4}, \frac{1}{4} e^{-i \text{angle}} \sin[\alpha] \right\}, \\ & \left\{ \frac{1}{4} e^{i \text{angle}} \sin[\alpha], \frac{\sin[\alpha]}{4}, \frac{1}{8} (1 + \sin[\alpha]^2), \frac{1}{8} e^{-i \text{angle}} (1 + \sin[\alpha]^2) \right\}, \\ & \left. \left\{ 0, 0, \frac{1}{8} e^{i \text{angle}} \cos[\alpha]^2, \frac{\cos[\alpha]^2}{8} \right\} \right\} \end{aligned}$$

In[53]:= rhoAknowingB = TraceSystem[projectedstate, {2}]

$$\begin{aligned} \text{Out[53]= } & \left\{ \left\{ \frac{\cos[\alpha]^2}{8} + \frac{1}{8} (1 + \sin[\alpha]^2), \frac{1}{4} e^{-i \text{angle}} \sin[\alpha] \right\}, \right. \\ & \left. \left\{ \frac{1}{4} e^{i \text{angle}} \sin[\alpha], \frac{\cos[\alpha]^2}{8} + \frac{1}{8} (1 + \sin[\alpha]^2) \right\} \right\} \end{aligned}$$

```
In[54]:= norm = FullSimplify[TraceSystem[projectedstate, {1, 2}]]
```

[simplifie complètement]

```
Out[54]= {{1/2}}
```

```
In[55]:= rhoKnowingBnorm = FullSimplify[1/norm[[1, 1]] rhoKnowingB]
```

[simplifie complètement]

```
Out[55]= {{1/2, 1/2 e^{-i angle Sin[alpha]}}, {1/2 e^{i angle Sin[alpha]}, 1/2}}
```

```
In[56]:= Tr[rhoKnowingBnorm.projector]
```

[trace tr]

```
Out[56]= 1/2 + Sin[alpha]/2
```


A.2.2 Analysis of the different protocols

Once again, a Matlab code is used to compare the different protocols:

```

1 % Joint probability
2 f = @(c_a,c_b,c_e,alpha,theta_a,theta_b,theta_e) 1/8*c_a*c_b*c_e
   *(1+cos(theta_a-theta_b)*sin(alpha)+cos(alpha).*(cos(theta_b-
   theta_e)+cos(theta_a-theta_e)*sin(alpha)));
3
4
5 % Selection of the protocol
6 BB84 = false;
7 noise = false;
8 australian = false;
9 GrosshansGrangier = true;
10
11 % Parameters depending on the protocol
12 if (noise)
13     AngleAlice = pi/2:pi:3*pi/2;
14     AngleBob = pi/4:pi/2:7*pi/4;
15     AngleEve = pi/2:pi:3*pi/2;
16     cAlice = 1;
17     cBob = 1/2;
18     cEve = 1;
19 end
20 if (BB84)
21     AngleAlice = 0:pi:pi;
22     AngleBob = 0:pi:pi;
23     AngleEve = 0:pi:pi;
24     cAlice = 1;
25     cBob = 1;
26     cEve = 1;
27 end
28 if (australian)
29     AngleAlice = pi/4:pi/2:7*pi/4;
30     AngleBob = pi/4:pi/2:7*pi/4;
31     AngleEve = pi/4:pi/2:7*pi/4;
32     cAlice = 1/2;
33     cBob = 1/2;
34     cEve = 1/2;
35 end
36 if (GrosshansGrangier)
37     AngleAlice = pi/4:pi/2:7*pi/4;
38     AngleBob = 0:pi:pi;
39     AngleEve = pi/4:pi/2:7*pi/4;
40     cAlice = 1/2;
41     cBob = 1;
42     cEve = 1/2;
43 end
44
45

```

```

46
47
48 jointproba = @(alpha , Alice , Bob , Eve) f(cAlice ,cBob ,cEve ,alpha ,
    Alice ,Bob ,Eve);
49
50 % Calculation of the different probabilities
51 proba_a_and_b = @(alpha , Alice , Bob) 0;
52 for i = 1:length(AngleEve)
53     proba_a_and_b = @(alpha , Alice , Bob) (proba_a_and_b(alpha ,
        Alice ,Bob)+jointproba(alpha , Alice ,Bob , AngleEve(i)));
54 end
55 proba_b_and_e = @(alpha , Bob , Eve) 0;
56 for j = 1:length(AngleAlice)
57     proba_b_and_e = @(alpha , Bob , Eve) (proba_b_and_e(alpha , Bob ,
        Eve)+jointproba(alpha , AngleAlice(j) , Bob , Eve));
58 end
59 proba_a = @(alpha , Alice) 0;
60 for l = 1:length(AngleBob)
61     proba_a = @(alpha , Alice) (proba_a(alpha , Alice)+
        proba_a_and_b(alpha , Alice , AngleBob(l)));
62 end
63 proba_b = @(alpha , Bob) 0;
64 for m = 1:length(AngleAlice)
65     proba_b = @(alpha , Bob) (proba_b(alpha , Bob)+proba_a_and_b(
        alpha , AngleAlice(m) , Bob));
66 end
67 proba_e = @(alpha , Eve) 0;
68 for n = 1:length(AngleBob)
69     proba_e = @(alpha , Eve) (proba_e(alpha , Eve)+proba_b_and_e(
        alpha , AngleBob(n) , Eve));
70 end
71
72 % Calculation of the entropies
73 H_A = @(alpha) 0;
74 for o = 1:length(AngleAlice)
75     H_A = @(alpha) (H_A(alpha)-proba_a(alpha , AngleAlice(o)).*
        log2(proba_a(alpha , AngleAlice(o))));
76 end
77 H_B = @(alpha) 0;
78 for p = 1:length(AngleBob)
79     H_B = @(alpha) (H_B(alpha)-proba_b(alpha , AngleBob(p)).*log2(
        proba_b(alpha , AngleBob(p))));
80 end
81 H_E = @(alpha) 0;
82 for q = 1:length(AngleEve)
83     H_E = @(alpha) (H_E(alpha)-proba_e(alpha , AngleEve(q)).*log2(
        proba_e(alpha , AngleEve(q))));
84 end
85 H_AB = @(alpha) 0;
86 for r = 1:length(AngleBob)

```

```

87     for s = 1:length(AngleAlice)
88         H_AB = @(alpha) (H_AB(alpha)-proba_a_and_b(alpha,
            AngleAlice(s),AngleBob(r)).*log2(proba_a_and_b(alpha,
            AngleAlice(s),AngleBob(r))));
89     end
90 end
91 H_BE = @(alpha) 0;
92 for t = 1:length(AngleEve)
93     for u = 1:length(AngleBob)
94         H_BE = @(alpha) (H_BE(alpha)-proba_b_and_e(alpha,
            AngleBob(u),AngleEve(t)).*log2(proba_b_and_e(alpha,
            AngleBob(u),AngleEve(t))));
95     end
96 end
97
98 % Calculation of the mutual information
99 I_AB = @(alpha) (H_A(alpha)+H_B(alpha)-H_AB(alpha));
100 I_BC = @(alpha) (H_B(alpha)+H_E(alpha)-H_BE(alpha));
101
102 % Plot
103 alpha = 0:(pi/200):pi/2;
104 QBER = 1-((1/2)+(sin(alpha)/2));
105 y = real(I_AB(alpha));
106 z = real(I_BC(alpha));
107 plot(QBER, y, 'r')
108 hold on;
109 plot(QBER, z, 'b')
110 grid on; grid minor;
111 plot(QBER, max(y-z, 0), 'g')
112 xlabel('QBER = 1-fidelity')
113 ylabel('Mutual information')
114 legend('I_A_B', 'I_B_E', '\Delta I = max(I_A_B-I_B_E, 0)')

```